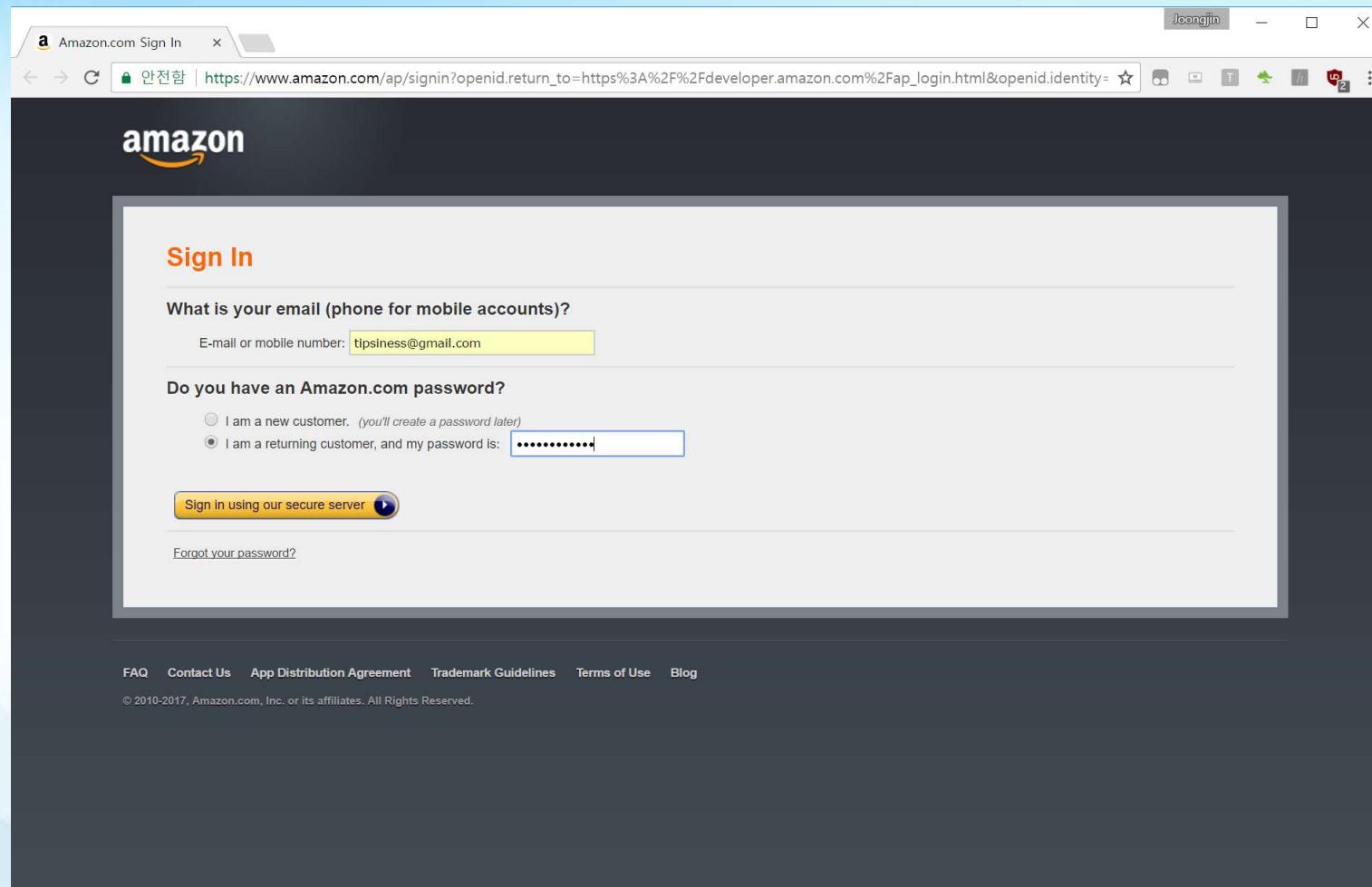


Third Party Cloud - Amazon Alexa

- Third Party Cloud Example (Amazon Alexa)

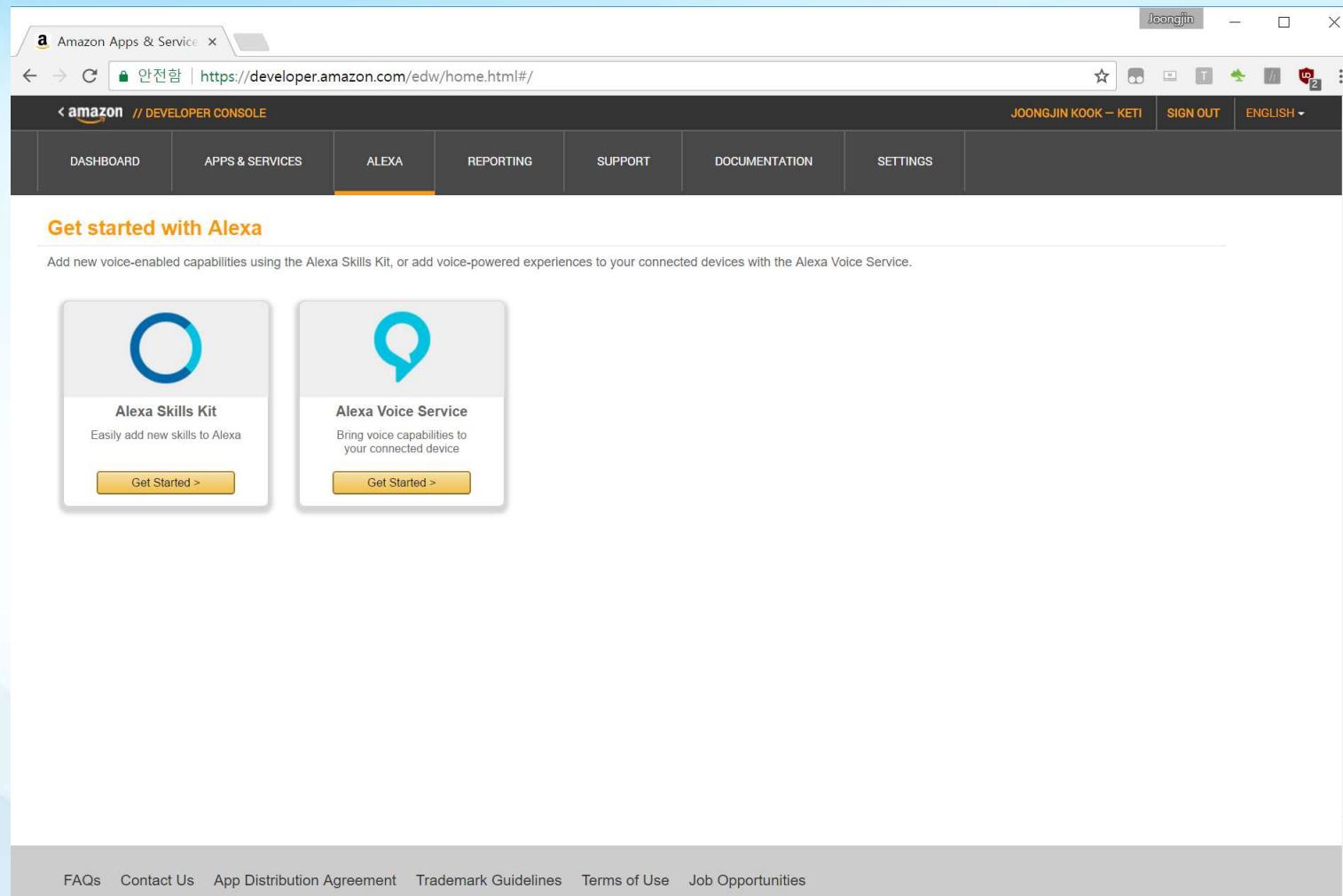
RPi Alexa

▶ Installation



RPi Alexa

▶ Installation



RPi Alexa

▶ Installation

The screenshot shows the Amazon Alexa Developer Console interface. The top navigation bar includes links for DASHBOARD, APPS & SERVICES, ALEXA (which is highlighted), REPORTING, SUPPORT, DOCUMENTATION, and SETTINGS. The user is signed in as JOONGJIN KOOK – KETI. A dropdown menu for 'Register a Product' is open, showing options for Device and Application. The main content area is titled 'Building with Alexa Voice Service'. It displays a table of registered products, with one entry visible:

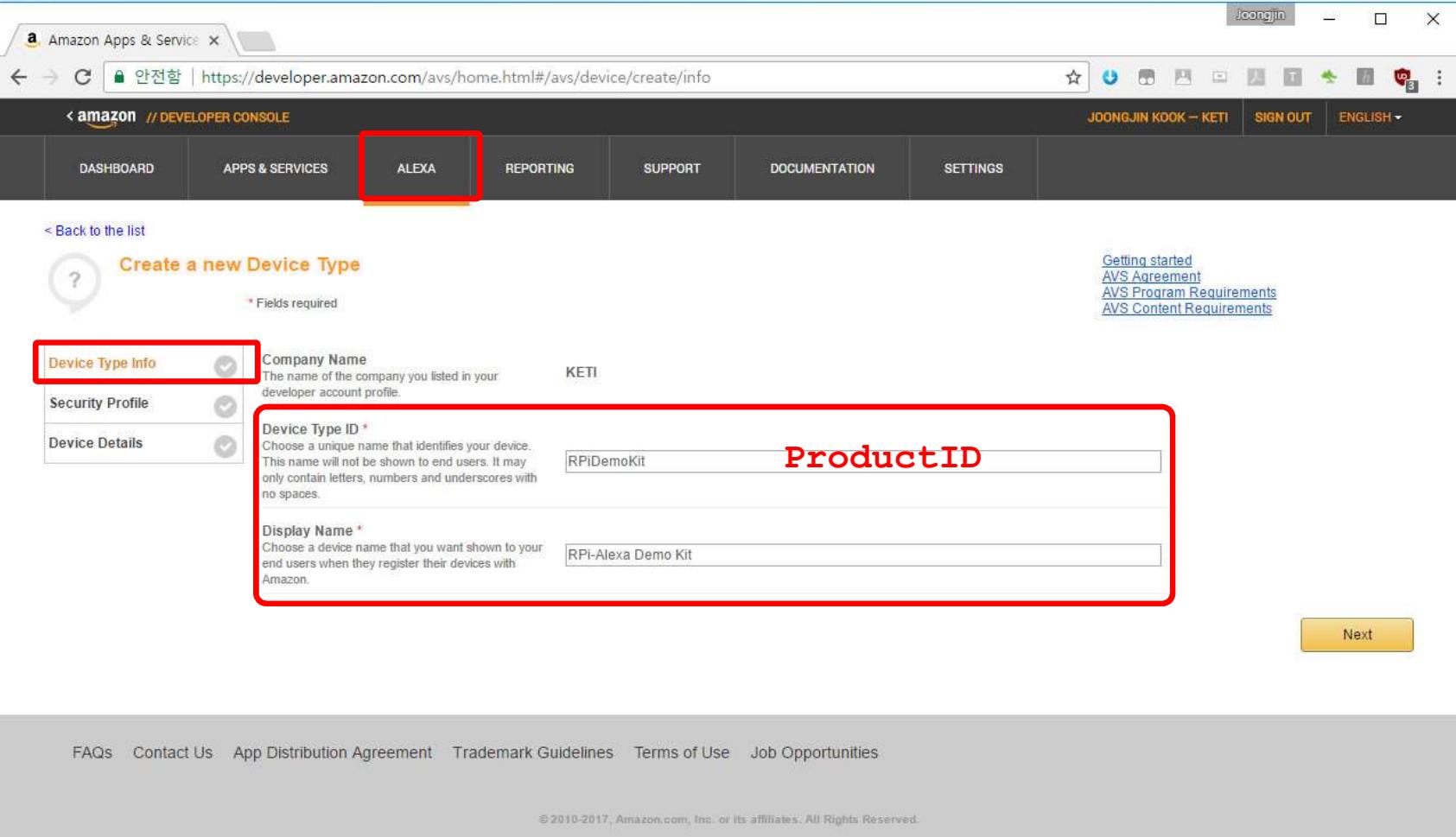
Product	Display Name	ID	Amazon ID	Type	Category	Actions
RPi Alexa Demo Kit	RPi Alexa Demo Kit	RPIDemoKit	A3ELQ80327C2B7	Device	Other	<button>Edit</button>

Below the table, there is a large orange button labeled 'AVS my_device'. At the bottom of the page, there are links for FAQs, Contact Us, App Distribution Agreement, Trademark Guidelines, Terms of Use, and Job Opportunities. The footer contains copyright information: © 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved.

<https://developer.amazon.com/avs/home.html>

RPi Alexa

▶ Installation



The screenshot shows the 'Create a new Device Type' page in the Amazon AVS Developer Console. The 'ALEXA' tab is selected in the navigation bar. The 'Device Type Info' section is highlighted with a red box. The 'ProductID' field contains 'RPIDemoKit' and is also highlighted with a red box. The 'Display Name' field contains 'RPI-Alexa Demo Kit'. A large red box encloses the 'ProductID' field and its associated input field. The 'Next' button is visible at the bottom right.

Amazon Apps & Services | <https://developer.amazon.com/avs/home.html#/avs/device/create/info>

amazon // DEVELOPER CONSOLE

JOONGJIN KOOK – KETI | SIGN OUT | ENGLISH

DASHBOARD APPS & SERVICES ALEXA REPORTING SUPPORT DOCUMENTATION SETTINGS

< Back to the list

Create a new Device Type

* Fields required

Device Type Info Company Name: KETI

Security Profile

Device Details

Device Type ID * ProductID: RPIDemoKit

Display Name * RPI-Alexa Demo Kit

Getting started
AVS Agreement
AVS Program Requirements
AVS Content Requirements

FAQs Contact Us App Distribution Agreement Trademark Guidelines Terms of Use Job Opportunities

© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved.

RPi Alexa

▶ Installation

The screenshot shows a web browser window titled "Amazon Apps & Service" with the URL <https://developer.amazon.com/avs/home.html#/avs/device/create/sp/general>. The page is titled "Create a new Device Type". A red box highlights the "Security Profile" section. The "Device Type Info" section has a green checkmark. The "Security Profile" section has a red box around it, and a red box also surrounds the "Create a new profile" button. The "Device Details" section has a red box around its checkbox. On the right, there are links for "Getting started", "AVS Agreement", "AVS Program Requirements", and "AVS Content Requirements". At the bottom, there are tabs for "General", "Web Settings", "Android/Kinect Settings", and "iOS Settings", with "General" selected. Below these tabs is a "Security Profile Description" field and a "Security Profile ID" field. At the very bottom is a "Next" button.

RPi Alexa

▶ Installation

The screenshot shows the 'Create a new Device Type' page in the Amazon Alexa Device Creation Wizard. The 'Security Profile' section is highlighted with a red box. The 'General' tab is selected. The security profile name is 'RPI-Alexa DemoKit Profile' and the description is 'RPI-Alexa DemoKit'. A 'Next' button is visible at the bottom right.

Create a new Device Type

* Fields required

Device Type Info ✓

Security Profile ? ✓

You need a security profile to identify your device. Your security profile credentials - client ID and client secret - allow your device to securely identify itself to the Alexa Voice Service. If you are building a website, click here to [Learn More](#). If you are building an Android or iOS app, click here to [Learn More](#).

Device Details ✓

Security Profile ? *
A security profile is how Amazon identifies your device.

General Web Settings Android/Kindle Settings iOS Settings

Security Profile Name *
Choose a name for your security profile.

Security Profile Description *
Choose a description for your security profile for Amazon services to use in communicating with you.

Next

FAQs Contact Us App Distribution Agreement Trademark Guidelines Terms of Use Job Opportunities

© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved.

RPi Alexa

▶ Installation

The screenshot shows a web browser window titled "Amazon Apps & Service" with the URL <https://developer.amazon.com/avs/home.html#/avs/device/create/sp/general>. The page is titled "Create Device" and displays a "Device Type Info" section with a green checkmark. Below it is a "Security Profile" section, also with a green checkmark, which is highlighted with a red rectangle. The "Device Details" section shows "RPI-Alexa DemoKit Profile" with an "Edit" button. Below these sections are tabs for "General", "Web Settings", "Android/Kindle Settings", and "iOS Settings". Under "General", there is a "Security Profile Description" field containing "RPI-Alexa DemoKit" and a "Security Profile ID" field containing "amzn1.application.5c9923488fb4cbeb40128eac56e7364". A large red rectangle highlights the "Client ID" and "Client Secret" fields. The "Client ID" field contains "amzn1.application-oa2-client.411ed70cc0ee486294c01c08951526a0" and the "Client Secret" field contains "4c9a99eb6f94d84ed7f5eb06ea41f1fa16b41531f089fa2f7035c6e4fc8131e5". A message at the bottom left says "Successfully created security profile." and a "Next" button is on the right. At the bottom, there are links for "FAQs", "Contact Us", "App Distribution Agreement", "Trademark Guidelines", "Terms of Use", and "Job Opportunities". The footer includes a copyright notice: "© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved."

RPi Alexa

▶ Installation

The screenshot shows the 'Create a new Device Type' page in the Amazon Developer Console. The 'ALEXA' tab is selected in the navigation bar. The 'Security Profile' section is highlighted with a red box. The 'Web Settings' tab is selected in the bottom navigation bar. A message indicates that a security profile is needed for device identification.

Create a new Device Type

* Fields required

Device Type Info ✓ You need a security profile to identify your device. Your security profile credentials - client ID and client secret - allow your device to securely identify itself to the Alexa Voice Service. If you are building a website, click here to [Learn More](#). If you are building an Android or iOS app, click here to [Learn More](#).

Security Profile ✓ Security Profile ? * A security profile is how Amazon identifies your device. Edit

Device Details ✓

General Web Settings Android/Kindle Settings iOS Settings

Allowed Origins ? Your website origin, when using Login with Amazon.

Allowed Return URLs ? If you make HTTPS calls to Login with Amazon with redirect_uris, specify them here.

Successfully created security profile.

Next

FAQs Contact Us App Distribution Agreement Trademark Guidelines Terms of Use Job Opportunities

© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved.

RPi Alexa

▶ Installation

The screenshot shows the 'Create a new Device Type' page in the Amazon Developer Console. The 'Security Profile' section is highlighted with a red box. The 'Web Settings' tab is selected. The 'Allowed Origins' field contains `https://localhost:3000`, and the 'Allowed Return URLs' field contains `https://localhost:3000/authresponse`. Both fields are also highlighted with red boxes.

Create a new Device Type

* Fields required

Device Type Info ✓ You need a security profile to identify your device. Your security profile credentials - client ID and client secret - allow your device to securely identify itself to the Alexa Voice Service. If you are building a website, click here to [Learn More](#). If you are building an Android or iOS app, click here to [Learn More](#).

Security Profile ✓

Device Details ✓

Cancel

General Web Settings Android/Kindle Settings iOS Settings

Allowed Origins `https://localhost:3000` Add Another

Allowed Return URLs `https://localhost:3000/authresponse` Add Another

Successfully created security profile.

Next

FAQs Contact Us App Distribution Agreement Trademark Guidelines Terms of Use Job Opportunities

© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved.

RPi Alexa

▶ Installation

The screenshot shows the Amazon Developer Console interface. The top navigation bar includes 'Amazon Apps & Services' and the user 'JOONGJIN KOOK - KETI'. The main menu has tabs for 'DASHBOARD', 'APPS & SERVICES', 'ALEXA' (which is highlighted), 'REPORTING', 'SUPPORT', 'DOCUMENTATION', and 'SETTINGS'. Below the menu, a banner says 'Building with Alexa Voice Service' with a 'Register a Product' button. A note below the banner states: 'For information on building your AVS product, see the [Getting started](#) guide. Review the [AVS Terms and Agreement](#), [Program Requirements](#) and [Content Requirements](#) prior to building your product.' The main content area displays a table of registered products. The table has columns: Product, Display Name, ID, Amazon ID, Type, Category, and Actions. One row is visible: 'RPi-Alexa Demo Kit' (Display Name), 'RPiDemoKit' (ID), 'A3ELQ80327C2B7' (Amazon ID), 'Device' (Type), 'Other' (Category), and an 'Edit' button (Actions). A red box highlights the 'Edit' button. To the left of the table, there's a small orange box labeled 'AVS my_device'. At the bottom of the page, there are links for 'FAQs', 'Contact Us', 'App Distribution Agreement', 'Trademark Guidelines', 'Terms of Use', and 'Job Opportunities'. A copyright notice at the very bottom reads: '© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved.'

Product	Display Name	ID	Amazon ID	Type	Category	Actions
	RPi-Alexa Demo Kit	RPiDemoKit	A3ELQ80327C2B7	Device	Other	Edit

RPi Alexa

▶ Installation

The screenshot shows a web browser window with two tabs open, both titled "Amazon Apps & Services". The URL in the address bar is <https://developer.amazon.com/lwa/sp/overview.html>. The browser title bar shows "joongjin". The main content area is the "DEVELOPER CONSOLE" for Amazon. The navigation bar includes links for DASHBOARD, APPS & SERVICES (which is highlighted), ALEXA, REPORTING, SUPPORT, DOCUMENTATION, and SETTINGS. Below the navigation bar, there are links for My Apps, App Testing Service, Promotions, Security Profiles, Login with Amazon, Dash Replenishment Service, Alexa News, GameCircle, and PC / Mac & Web Instant Access. There are also links for Tester Management, Advertise Your App, and Mobile Ads.

A modal dialog box is centered on the screen with the title "Enter Consent Screen Information". The dialog contains the following text: "Login with Amazon requires additional information that will be shown to users whenever you request access to their personal data." It also states, "The user can choose to share some personal data in their Amazon Profile." Below this text, there is a field labeled "Consent Privacy Notice URL" with the value "http://example.com". There is also a field labeled "Consent Logo Image" with a placeholder "UPLOAD IMAGE" and a red rectangular box drawn around it. At the bottom of the dialog are "Cancel" and "Save" buttons.

At the bottom of the page, there are links for FAQs, Contact Us, App Distribution Agreement, Trademark Guidelines, Terms of Use, and Job Opportunities. A copyright notice at the very bottom reads "© 2010-2017, Amazon.com, Inc. or its affiliates. All Rights Reserved."

RPi Alexa

▶ Installation

The screenshot shows the Amazon Developer Console interface. At the top, there are two tabs labeled "Amazon Apps & Services". The main navigation bar includes links for DASHBOARD, APPS & SERVICES (which is highlighted with a red box), ALEXA, REPORTING, SUPPORT, DOCUMENTATION, and SETTINGS. Below the navigation bar, there are several service links: My Apps, App Testing Service, Promotions, Security Profiles, Login with Amazon (which is orange), Dash Replenishment Service, Alexa New, GameCircle, and PC / Mac & Web Instant Access. Under the "Security Profiles" link, there are links for Tester Management, Advertise Your App, and Mobile Ads.

Login with Amazon

Login with Amazon allows users to login to registered third party websites or apps ("clients") using their Amazon user name and password. Clients may ask the user to share some personal information from their Amazon profile, including name, email address, and zip code. To get started, select an existing Security Profile or create a new Security Profile. [Learn More](#)

[Create a New Security Profile](#)

✓ Login with Amazon successfully enabled for Security Profile. Click  to manage Security Profile.

Login with Amazon Configurations

Security Profile Name	OAuth2 Credentials	Manage
RPi-Alexa DemoKit Profile	Show Client ID and Client Secret	

RPi Alexa

▶ Installation

The screenshot shows the Amazon Developer Console interface. At the top, there are two tabs labeled "Amazon Apps & Services". The main navigation bar includes links for DASHBOARD, APPS & SERVICES (which is highlighted with a red box), ALEXA, REPORTING, SUPPORT, DOCUMENTATION, and SETTINGS. Below the navigation bar, there are several service links: My Apps, App Testing Service, Promotions, Security Profiles (which is also highlighted with a red box), Login with Amazon, Dash Replenishment Service, Alexa New, GameCircle, PC / Mac & Web Instant Access, Tester Management, Advertise Your App, and Mobile Ads.

In the center, there is a "Login with Amazon" section with a "Create a New Security Profile" button. A green success message states: "✓ Login with Amazon successfully enabled for Security Profile. Click ⚙ to manage Security Profile." Below this, there is a table titled "Login with Amazon Configurations". It has columns for "Security Profile Name", "OAuth2 Credentials", and "Manage". A row in the table shows the "Security Profile Name" as "RPI-Alexa DemoKit Profile" and the "OAuth2 Credentials" as "Client ID: amzn1.application-
oa2-client.411ed70cc0ee486294c01c08951526a0
Client Secret: 4c9a99eb6f94d84ed7f5eb06ea41f1fa16b41531f089fa2f7035c6e4fc8131e5". This row is also highlighted with a red box.

RPi Alexa

► Download Source

► \$ git clone https://github.com/alexa/alexa-avs-sample-app.git

► Installation

► \$ cd alexa/alexa-avs-sample-app

► Edit Install Script

► \$ nano automated_install.sh

→ add your ProductID, ClientID, and ClientSecret

```
# This is the name given to your device or mobile app in the Amazon developer portal. To look this up, navigate to https://developer.amazon.com/edw/home.html. It may be labeled Device Type ID.  
ProductID=RPiDemoKit  
  
# Retrieve your client ID from the web settings tab within the developer console : https://developer.amazon.com/edw/home.html  
ClientID=amzn1.application-oa2-client.411ed70cc0ee486294c01c08951526a0  
  
# Retrieve your client secret from the web settings tab within the developer console: https://developer.amazon.com/edw/home.html  
ClientSecret=4c9a99eb6f94d84ed7f5eb06ea41f1fa16b41531f089fa2f7035c6e4fc8131e5  
  
#-----  
# No need to change anything below this...
```

RPi Alexa

▶ Installation

- ▶ \$ cd alexa/alexavs-sample-app
- ▶ \$ chmod +x automated_install.sh
- ▶ \$./automated_install.sh

```
pi@raspberrypi:~/Alexa/alexavs-sample-app $ ls  
automated_install.sh  Installer_Licenses.txt  NOTICE.txt  RELEASE.txt  
CONTRIBUTING.md       LICENSE.txt           README.md    samples  
pi@raspberrypi:~/Alexa/alexavs-sample-app $ chmod +x automated_install.sh  
pi@raspberrypi:~/Alexa/alexavs-sample-app $  
pi@raspberrypi:~/Alexa/alexavs-sample-app $ ls -l automated_install.sh  
-rwxr-xr-x 1 pi pi 19842 Jun  6 15:54 automated_install.sh  
pi@raspberrypi:~/Alexa/alexavs-sample-app $  
pi@raspberrypi:~/Alexa/alexavs-sample-app $ █
```

ARTIK 710 based Alexa

▶ Install Dependencies

- ▶ # dnf update
- ▶ # dnf install git-core
- ▶ # dnf install vlc
- ▶ Node.js
- ▶ JDK

▶ Download Source

- ▶ \$ git clone https://github.com/alexa/alexa-avs-sample-app.git

▶ Installation

- ▶ \$ cd alexa/alexa-avs-sample-app

▶ Edit Install Script

- ▶ \$ nano automated_install.sh
- add your ProductID, ClientID, and ClientSecret

ARTIK 710 based Alexa

▶ Install VLC

- ▶ # dnf install https://download1.rpmfusion.org/free/fedora/rpmfusion-free-release-\$(rpm -E %fedora).noarch.rpm
- ▶ # dnf install vlc

```
[root@localhost ~]# dnf install https://download1.rpmfusion.org/free/fedora/rpmfusion-free-release-$(rpm -E %fedora).noarch.rpm
Last metadata expiration check: 1:09:50 ago on Sat Jul 22 04:03:40 2017.
Package rpmfusion-free-release-24-3.noarch is already installed, skipping.
Dependencies resolved.
Nothing to do.
Complete!
[root@localhost ~]# dnf install vlc
Last metadata expiration check: 1:10:37 ago on Sat Jul 22 04:03:40 2017.
Dependencies resolved.
=====
Package           Arch    Version      Repository      Size
=====
Installing:
SDL_image        armv7hl 1.2.12-14.fc24      fedora          44 k
dejavu-sans-fonts noarch   2.35-3.fc24       fedora         1.5 M
dejavu-serif-fonts noarch   2.35-3.fc24       fedora         804 k
desktop-file-utils armv7hl 0.23-2.fc24      updates         71 k
fluidsynth-libs   armv7hl 1.1.6-7.fc24      fedora          209 k
freerdp-libs     armv7hl 2:2.0.0-5.git.b02943a.fc24 fedora         797 k
ftgl              armv7hl 2.1.3-0.14.rc5.fc24   fedora         102 k
game-music-emu   armv7hl 0.6.1-1.fc24      updates         143 k
glibmm24          armv7hl 2.48.1-1.fc24      fedora         451 k
glx-utils         armv7hl 8.3.0-3.fc24      fedora          36 k
jack-audio-connection-kit

```

RPi Alexa

► Setup Environment Variables

- ▶ LD_LIBRARY_PATH: /usr/lib/vlc
- ▶ \$ export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib/vlc
- ▶ VLC_PLUGIN_PATH: /usr/lib/vlc/plugins
- ▶ \$ export VLC_PLUGIN_PATH=/usr/lib/vlc/plugins

```
[root@localhost ~]# export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/lib/vlc
[root@localhost ~]# export VLC_PLUGIN_PATH=/usr/lib/vlc/plugins
[root@localhost ~]#
[root@localhost ~]# █
```

RPi Alexa

▶ Install JDK

- ▶ URL - <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

The screenshot shows a web browser window titled "Java SE Development Kit 8 Downloads" from the Oracle website. The URL in the address bar is <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>. The page content includes a sidebar with links to Java SE, Java EE, Java ME, Java Support, Java Advanced & Suite, Java Embedded, Java DB, Web Tier, Java Card, Java TV, New to Java, Community, and Java Magazine. The main content area features tabs for Overview, Downloads, Documentation, Community, Technologies, and Training. A section titled "Java SE Development Kit 8 Downloads" explains the purpose of the JDK and lists tools for development and testing. Below this, a "See also:" section provides links to developer newsletters, Java Developer Day workshops, and Java Magazine. A "JDK 8u141 checksum" link is also present. A prominent call-to-action box states: "You must accept the Oracle Binary Code License Agreement for Java SE to download this software." and "Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software." A table titled "Java SE Development Kit 8u141" lists available downloads for various platforms, including Linux ARM 32 Hard Float ABI, Linux ARM 64 Hard Float ABI, Linux x86, Linux x86_64, Mac OS X, Solaris SPARC 64-bit, Solaris SPARC 64-bit, Solaris x64, Solaris x64, Windows x86, and Windows x64. Each entry includes the file size and a download link.

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.88 MB	jdk-8u141-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.83 MB	jdk-8u141-linux-arm64-vfp-hflt.tar.gz
Linux x86	164.66 MB	jdk-8u141-linux-i586.rpm
Linux x86	179.4 MB	jdk-8u141-linux-i586.tar.gz
Linux x64	162.11 MB	jdk-8u141-linux-x64.rpm
Linux x64	176.92 MB	jdk-8u141-linux-x64.tar.gz
Mac OS X	226.6 MB	jdk-8u141-macosx-x64.dmg
Solaris SPARC 64-bit	139.84 MB	jdk-8u141-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.17 MB	jdk-8u141-solaris-sparcv9.tar.gz
Solaris x64	140.59 MB	jdk-8u141-solaris-x64.tar.Z
Solaris x64	97.01 MB	jdk-8u141-solaris-x64.tar.gz
Windows x86	190.95 MB	jdk-8u141-windows-i586.exe
Windows x64	197.78 MB	jdk-8u141-windows-x64.exe

RPi Alexa

▶ Install JDK

- ▶ URL - <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

```
[root@localhost ~]# ls
alexa-avs-sample-app  jdk-8u141-linux-arm64-vfp-hflt.tar.gz
anaconda-ks.cfg        mosquitto-1.4.13-1.fc24.armv7hl.rpm
jdk1.8.0_141
[root@localhost ~]#
[root@localhost ~]# export JAVA_HOME=/root/jdk1.8.0_141
[root@localhost ~]# export PATH=$JAVA_HOME/bin:$PATH
[root@localhost ~]#
[root@localhost ~]#
```

RPi Alexa

▶ Install Maven

- ▶ URL - <http://mirror.navercorp.com/apache/maven/maven-3/3.5.0/binaries/apache-maven-3.5.0-bin.tar.gz>

```
[root@localhost ~]# wget http://mirror.navercorp.com/apache/maven/maven-3/3.5.0/
binaries/apache-maven-3.5.0-bin.tar.gz
--2017-07-23 05:31:59--  http://mirror.navercorp.com/apache/maven/maven-3/3.5.0/
binaries/apache-maven-3.5.0-bin.tar.gz
Resolving mirror.navercorp.com (mirror.navercorp.com)... 125.209.216.167
Connecting to mirror.navercorp.com (mirror.navercorp.com)|125.209.216.167|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 8534562 (8.1M) [application/x-gzip]
Saving to: 'apache-maven-3.5.0-bin.tar.gz'

apache-maven-3.5.0- 100%[=====] 8.14M 3.61MB/s   in 2.3s

2017-07-23 05:32:01 (3.61 MB/s) - 'apache-maven-3.5.0-bin.tar.gz' saved [8534562
/8534562]

[root@localhost ~]# ls -l apache-maven-3.5.0-bin.tar.gz
-rw-r--r-- 1 root root 8534562 Apr  7 04:29 apache-maven-3.5.0-bin.tar.gz
[root@localhost ~]#
[root@localhost ~]# tar xfz apache-maven-3.5.0-bin.tar.gz
[root@localhost ~]#
```

RPi Alexa

▶ Install Maven

- ▶ URL - <http://mirror.navercorp.com/apache/maven/maven-3/3.5.0/binaries/apache-maven-3.5.0-bin.tar.gz>

```
[root@localhost ~]# ls
alexa-avs-sample-app          jdk1.8.0_141
anaconda-ks.cfg                jdk-8u141-linux-arm64-vfp-hflt.tar.gz
apache-maven-3.5.0             mosquitto-1.4.13-1.fc24.armv7hl.rpm
apache-maven-3.5.0-bin.tar.gz
[root@localhost ~]#
[root@localhost ~]# export PATH=/root/apache-maven-3.5.0/bin:$PATH
[root@localhost ~]#
[root@localhost ~]# █
```

RPi Alexa

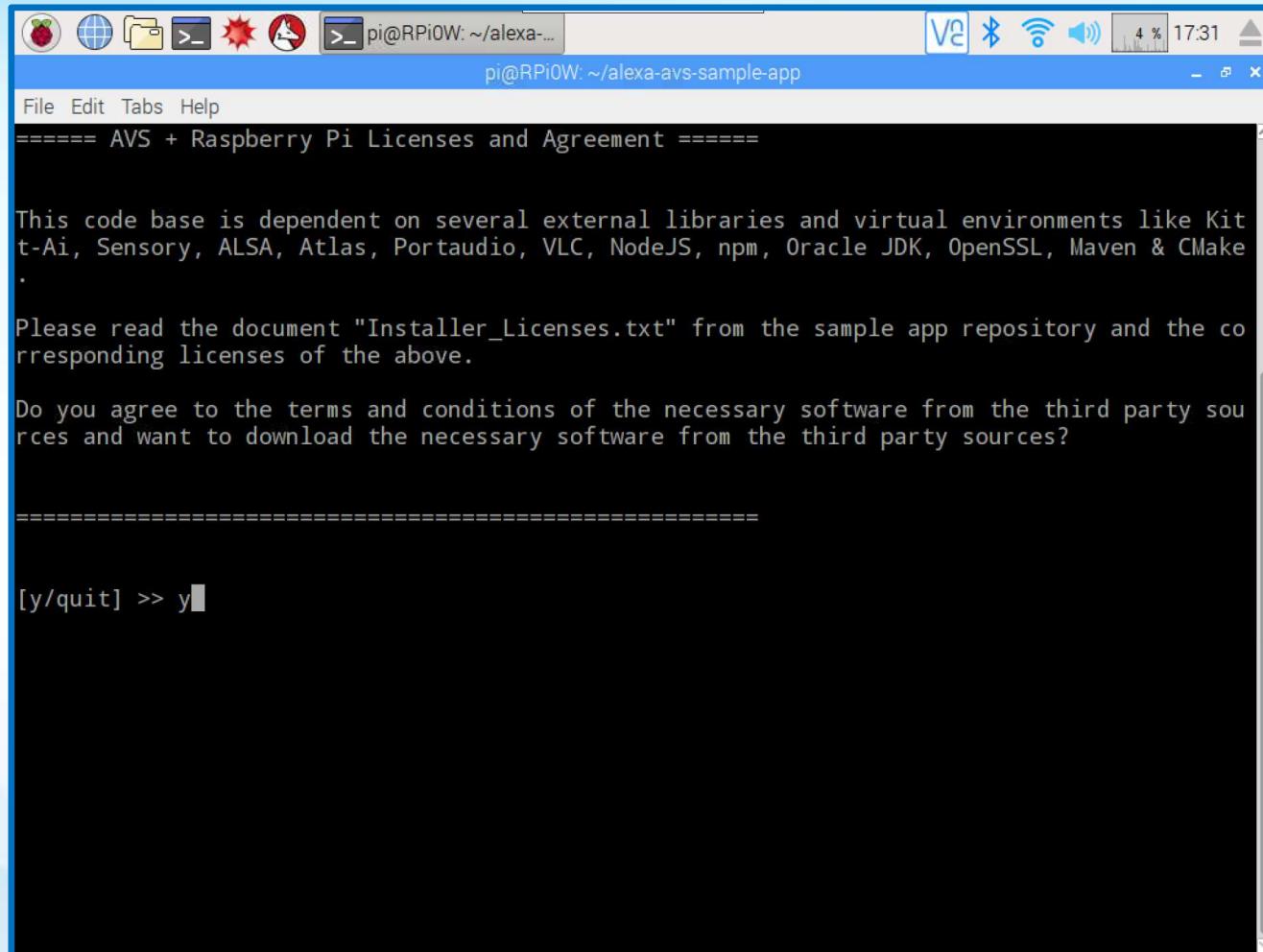
▶ Installation

- ▶ \$ cd alexa/alexavs-sample-app
- ▶ \$ chmod +x automated_install.sh
- ▶ \$./automated_install.sh

```
pi@raspberrypi:~/Alexa/alexavs-sample-app $ ls  
automated_install.sh  Installer_Licenses.txt  NOTICE.txt  RELEASE.txt  
CONTRIBUTING.md       LICENSE.txt           README.md    samples  
pi@raspberrypi:~/Alexa/alexavs-sample-app $ chmod +x automated_install.sh  
pi@raspberrypi:~/Alexa/alexavs-sample-app $  
pi@raspberrypi:~/Alexa/alexavs-sample-app $ ls -l automated_install.sh  
-rwxr-xr-x 1 pi pi 19842 Jun  6 15:54 automated_install.sh  
pi@raspberrypi:~/Alexa/alexavs-sample-app $  
pi@raspberrypi:~/Alexa/alexavs-sample-app $ █
```

RPi Alexa

▶ Installation



A screenshot of a terminal window titled "pi@RPi0W: ~/alexa-...". The window contains the following text:

```
pi@RPi0W: ~/alexa-avc-sample-app
File Edit Tabs Help
===== AVS + Raspberry Pi Licenses and Agreement =====

This code base is dependent on several external libraries and virtual environments like Kit
t-Ai, Sensory, ALSA, Atlas, Portaudio, VLC, NodeJS, npm, Oracle JDK, OpenSSL, Maven & CMake
.

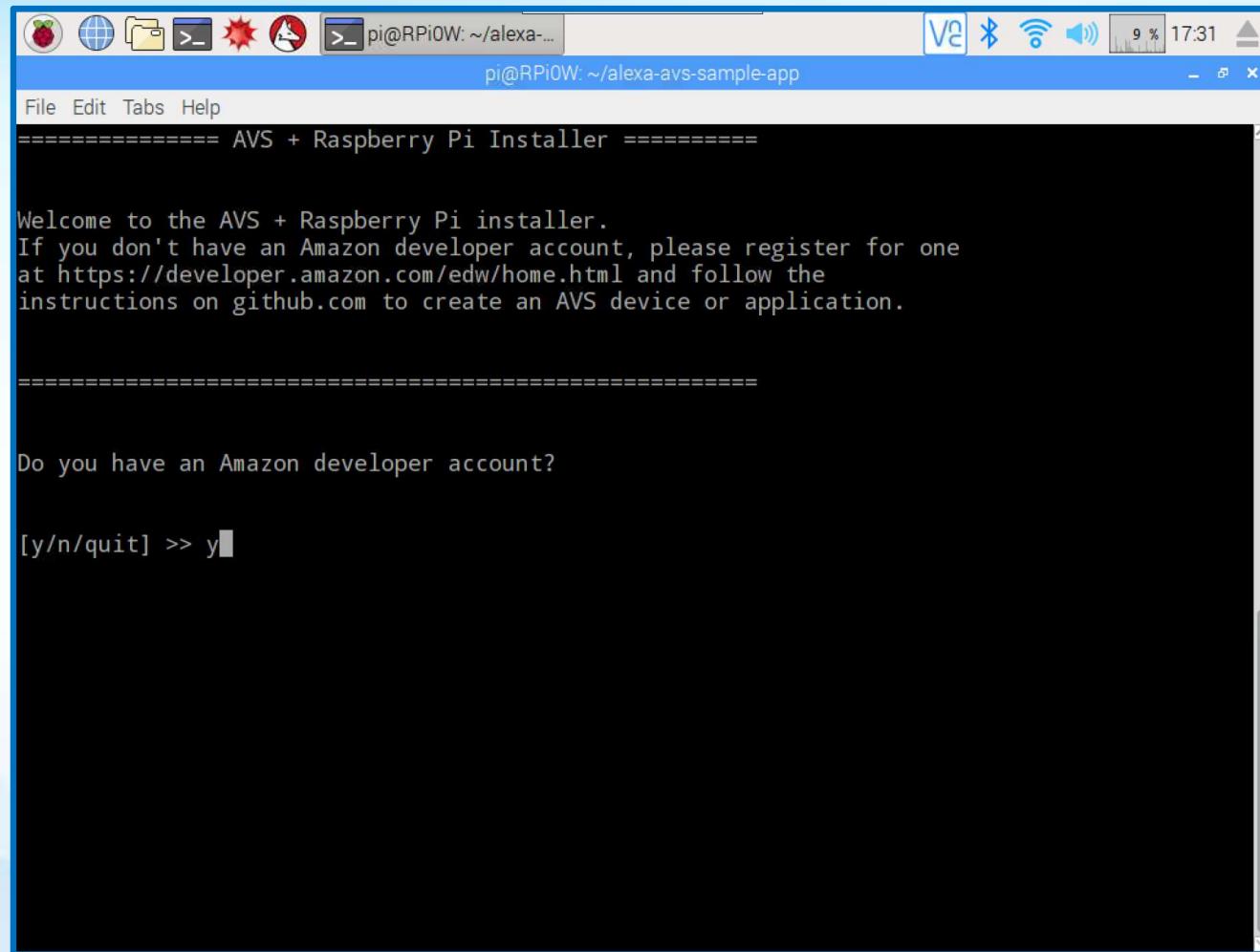
Please read the document "Installer_Licenses.txt" from the sample app repository and the co
rresponding licenses of the above.

Do you agree to the terms and conditions of the necessary software from the third party sou
rces and want to download the necessary software from the third party sources?

=====
[y/quit] >> y
```

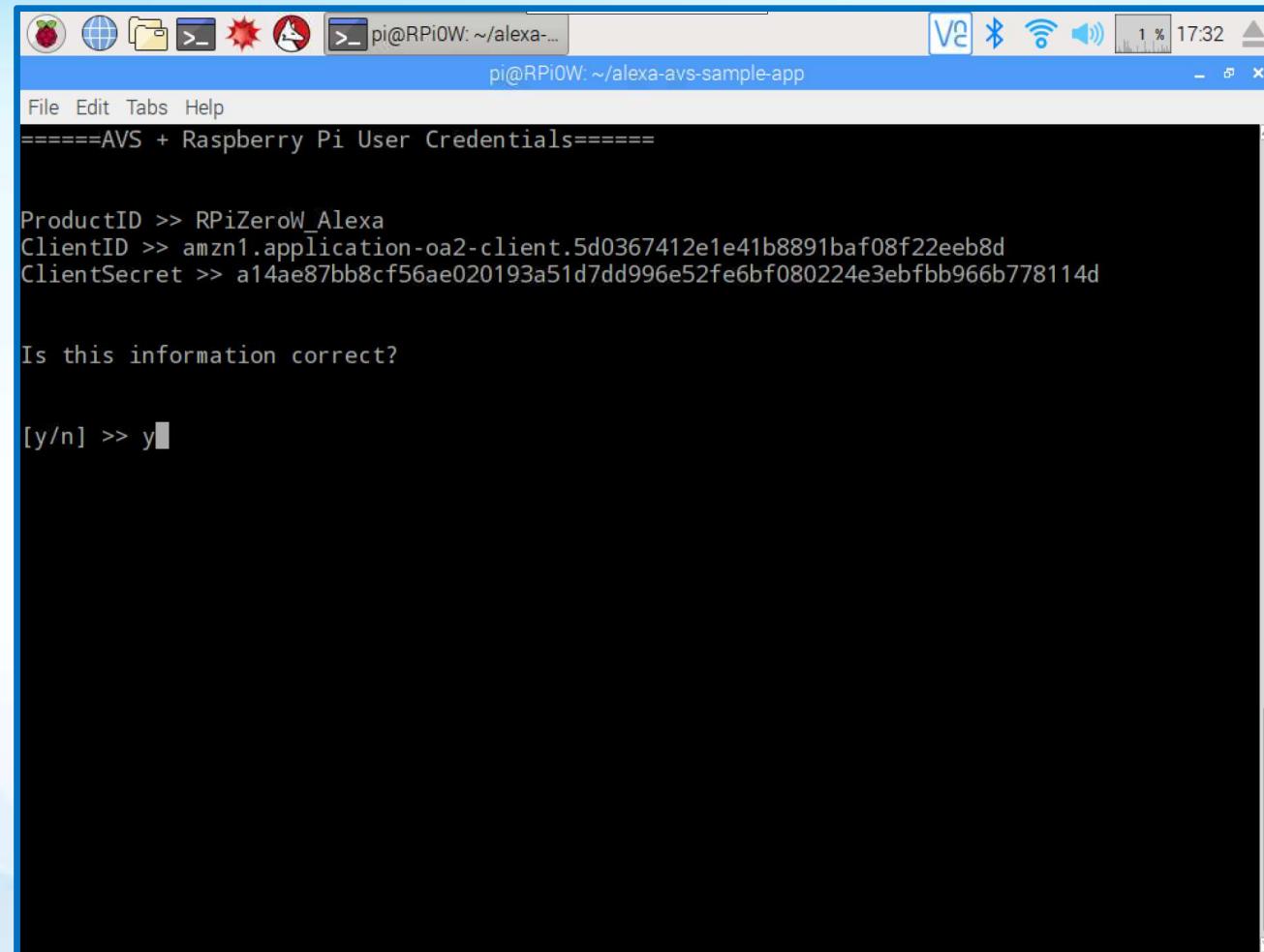
RPi Alexa

▶ Installation



RPi Alexa

▶ Installation

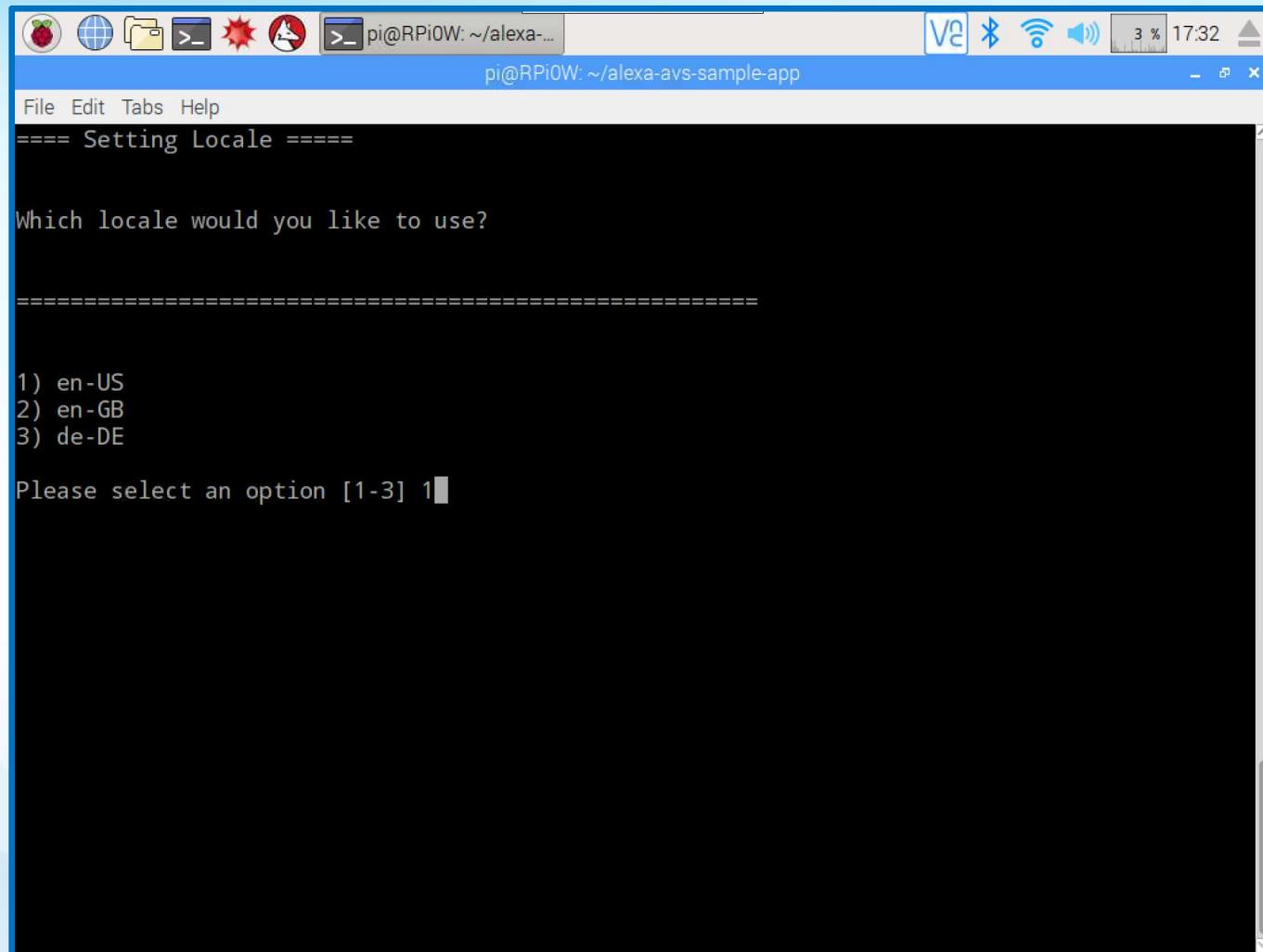


A screenshot of a terminal window titled "pi@RPi0W: ~/alexa-...". The window shows the following text:

```
File Edit Tabs Help  
=====AVS + Raspberry Pi User Credentials=====  
  
ProductID >> RPiZeroW_Alexa  
ClientID >> amzn1.application-oa2-client.5d0367412e1e41b8891baf08f22eeb8d  
ClientSecret >> a14ae87bb8cf56ae020193a51d7dd996e52fe6bf080224e3ebfbb966b778114d  
  
Is this information correct?  
  
[y/n] >> y
```

RPi Alexa

▶ Installation

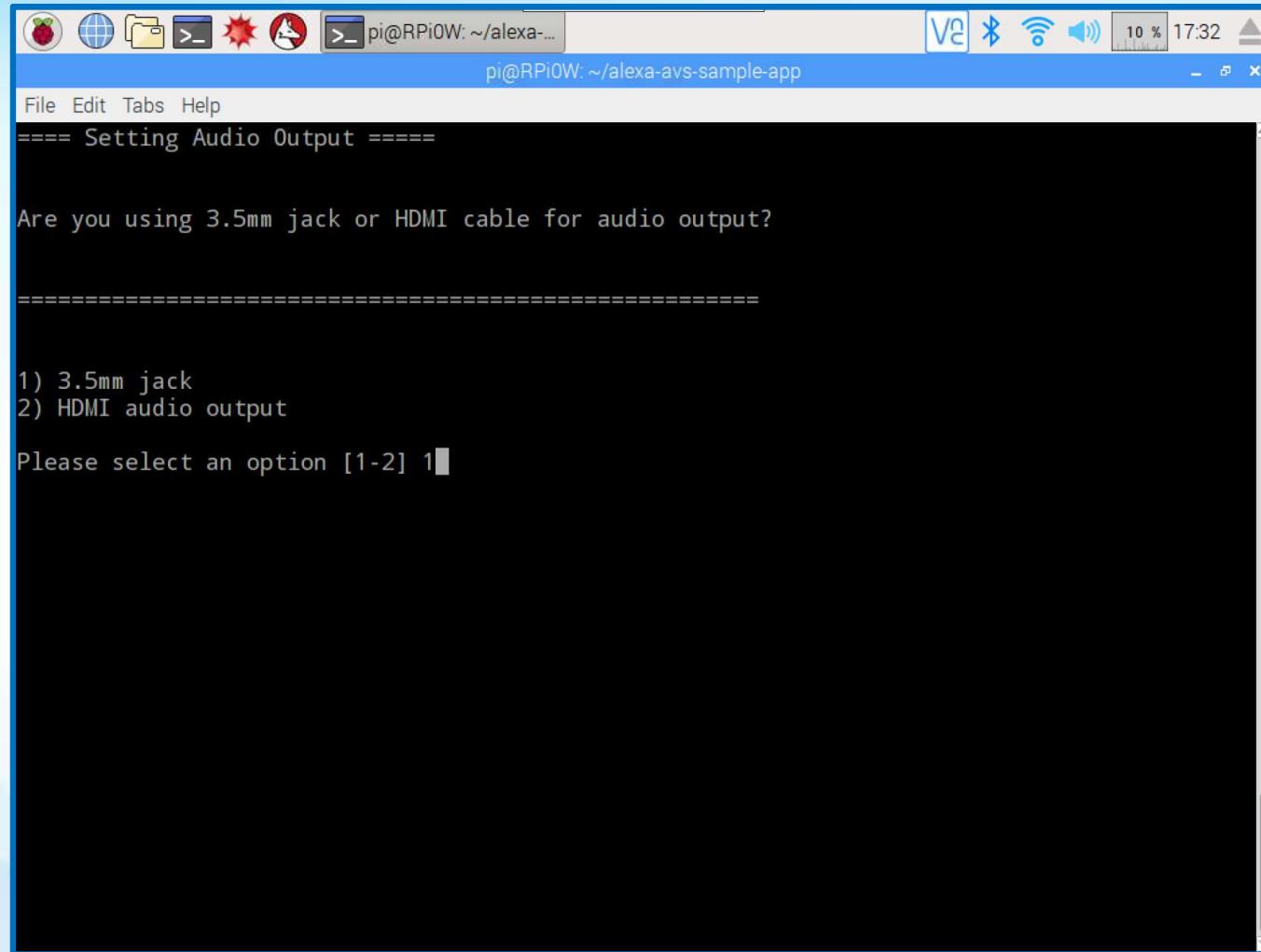


A screenshot of a terminal window titled "pi@RPi0W: ~/alexa-av...". The window shows a command-line interface for setting the locale. The text in the terminal is as follows:

```
File Edit Tabs Help
==== Setting Locale ====
Which locale would you like to use?
=====
1) en-US
2) en-GB
3) de-DE
Please select an option [1-3] 1
```

RPi Alexa

▶ Installation

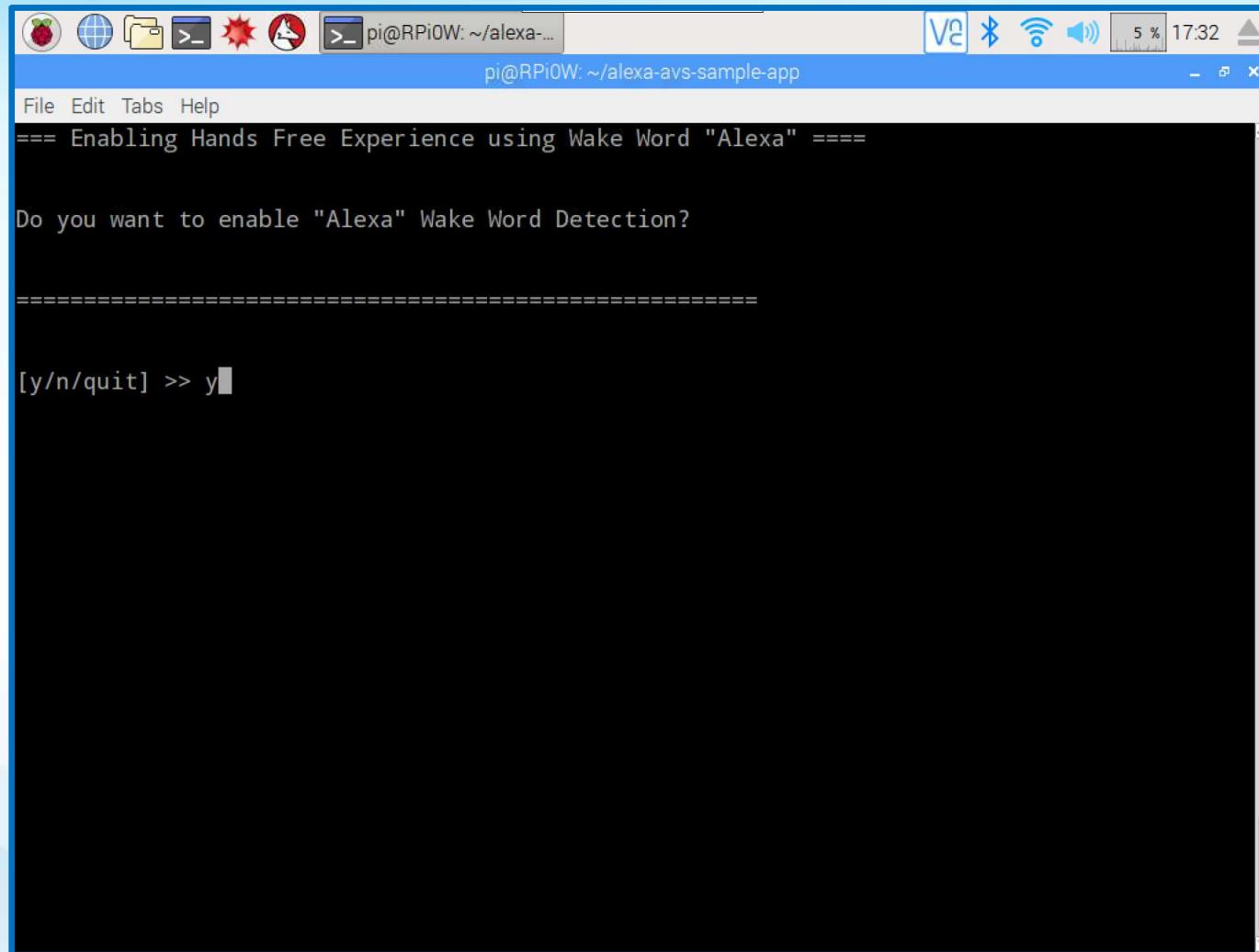


A screenshot of a terminal window on a Raspberry Pi. The window title is "pi@RPi0W: ~/alexa-...". The terminal displays the following text:

```
File Edit Tabs Help
==== Setting Audio Output ====
Are you using 3.5mm jack or HDMI cable for audio output?
=====
1) 3.5mm jack
2) HDMI audio output
Please select an option [1-2] 1
```

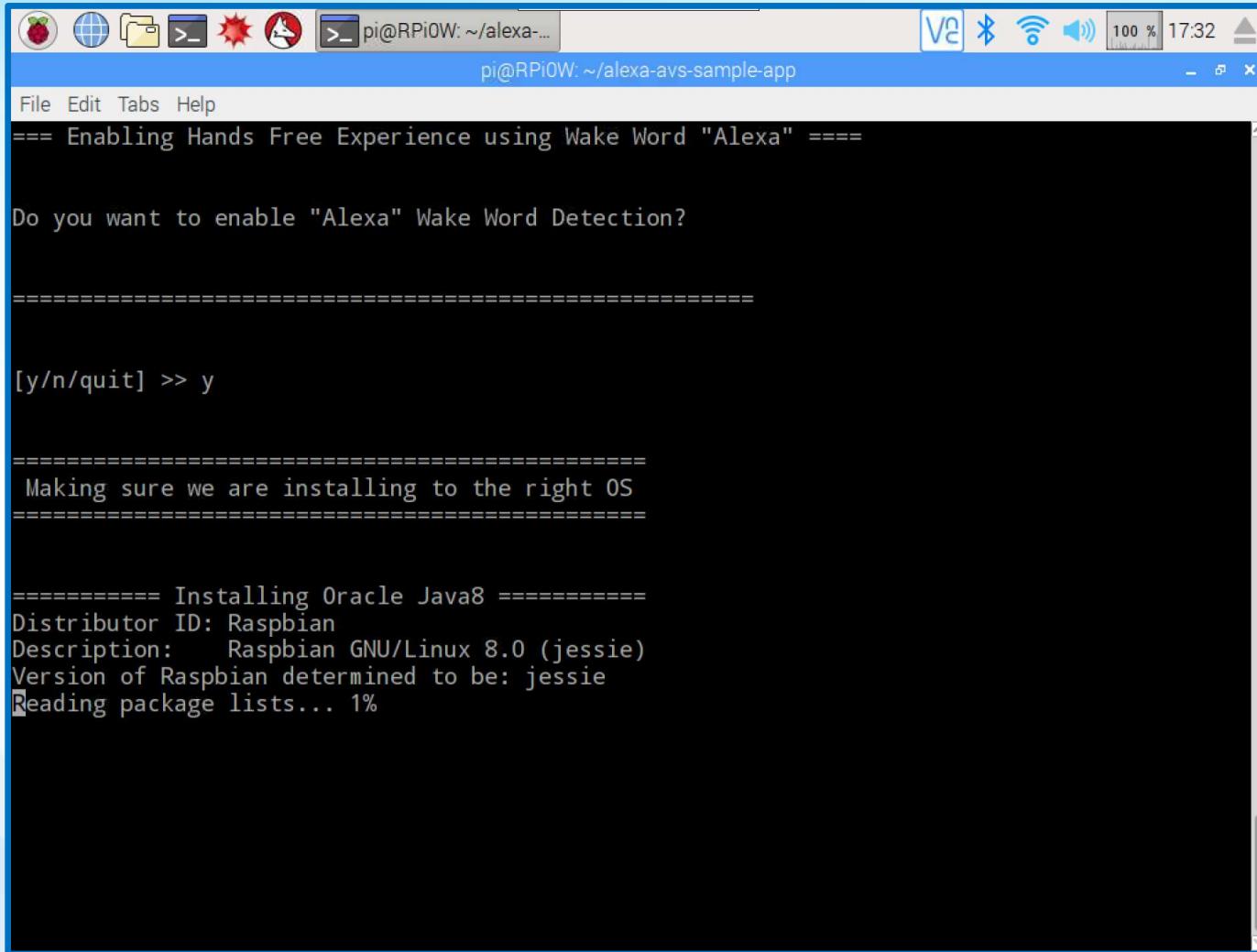
RPi Alexa

▶ Installation



RPi Alexa

▶ Installation

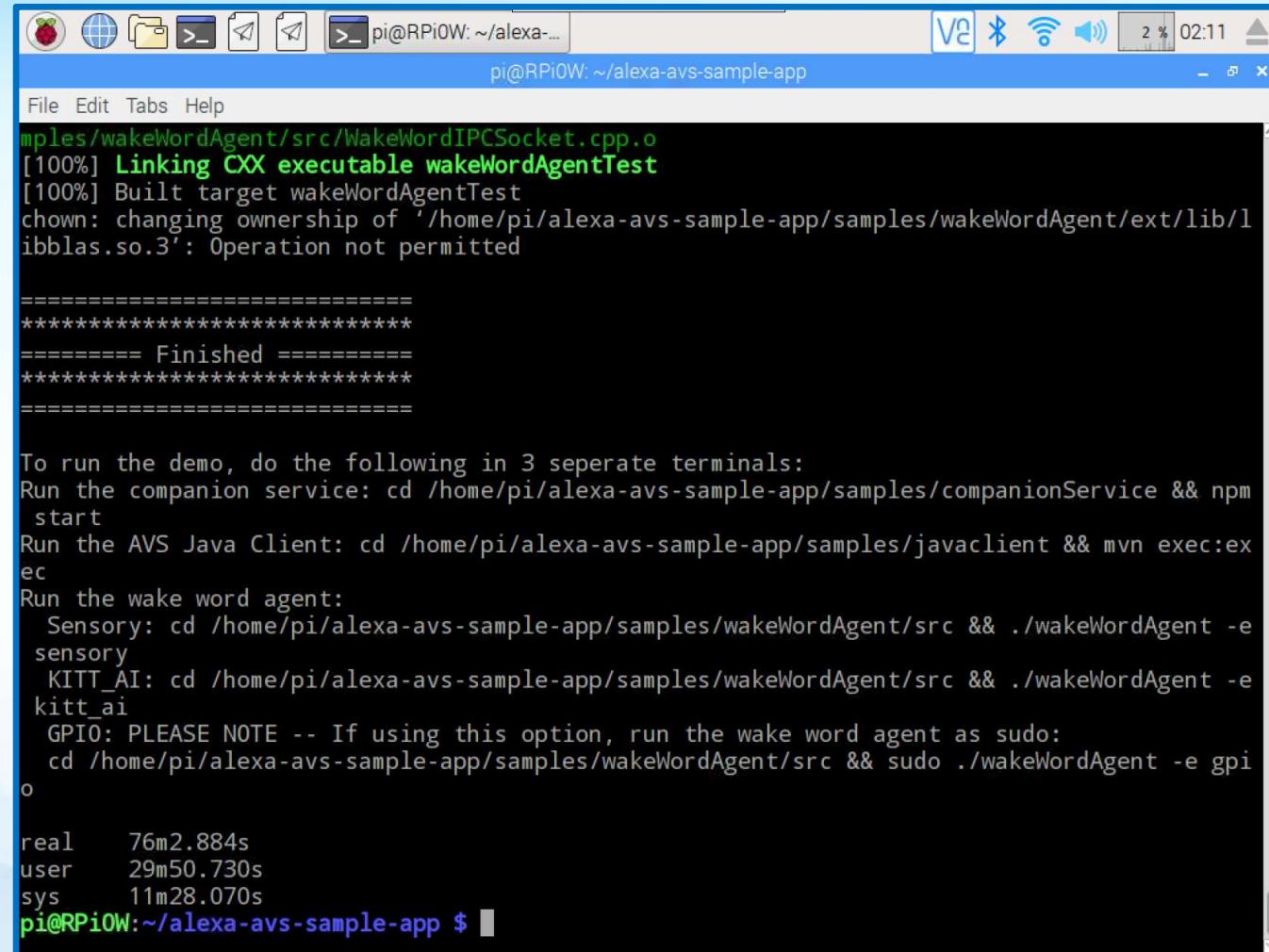


The screenshot shows a terminal window titled "pi@RPi0W: ~/alexa-...". The window contains the following text:

```
File Edit Tabs Help
==== Enabling Hands Free Experience using Wake Word "Alexa" ====
Do you want to enable "Alexa" Wake Word Detection?
=====
[y/n/quit] >> y
=====
Making sure we are installing to the right OS
=====
===== Installing Oracle Java8 =====
Distributor ID: Raspbian
Description:    Raspbian GNU/Linux 8.0 (jessie)
Version of Raspbian determined to be: jessie
Reading package lists... 1%
```

RPi Alexa

► Installation Completed



The screenshot shows a terminal window titled "pi@RPi0W: ~/alexa-av...". The window contains the following text:

```
File Edit Tabs Help
samples/wakeWordAgent/src/WakeWordIPCSocket.cpp.o
[100%] Linking CXX executable wakeWordAgentTest
[100%] Built target wakeWordAgentTest
chown: changing ownership of '/home/pi/alexa-avs-sample-app/samples/wakeWordAgent/ext/lib/libblas.so.3': Operation not permitted

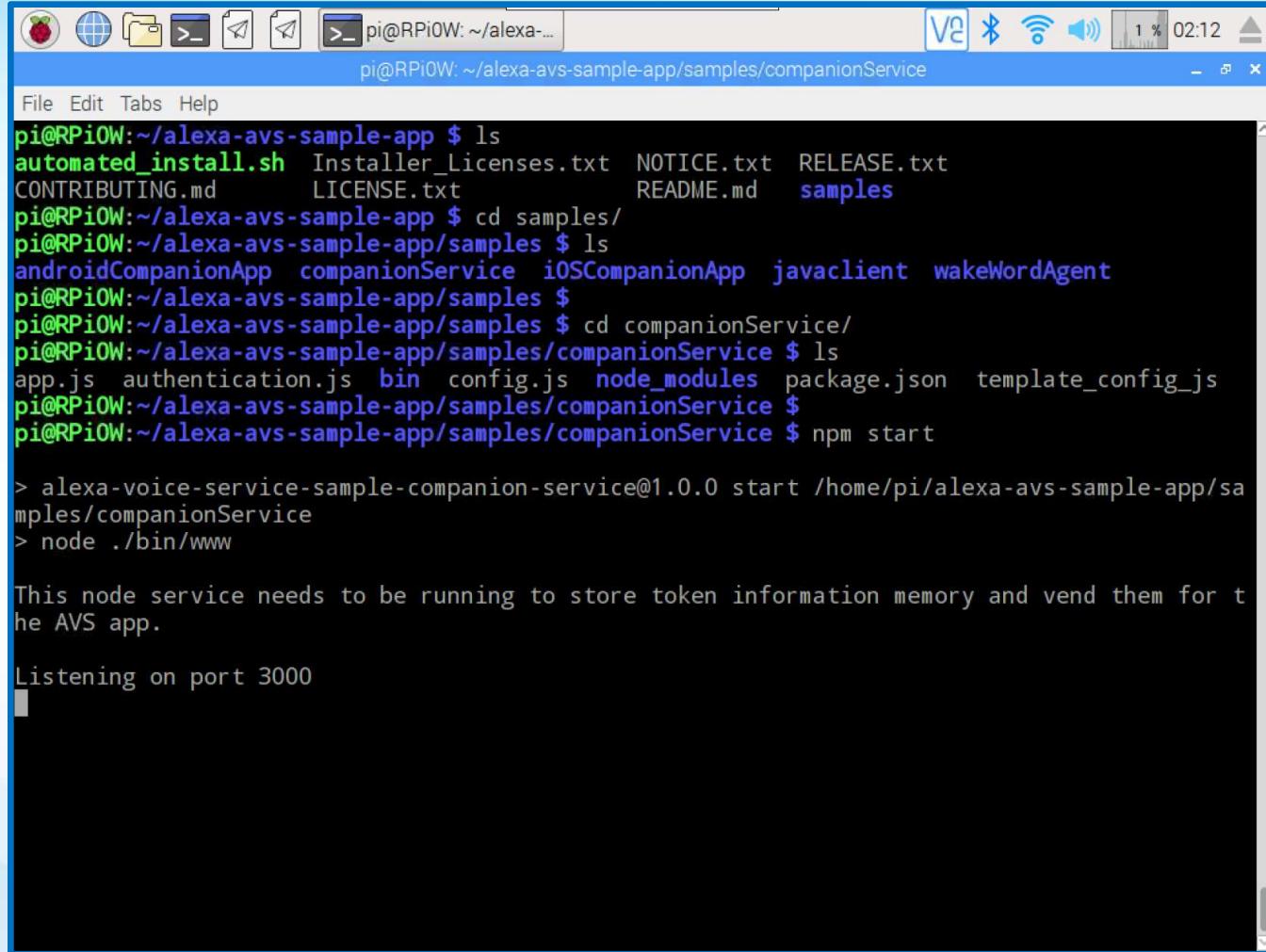
=====
*****
===== Finished =====
*****
=====
```

To run the demo, do the following in 3 seperate terminals:
Run the companion service: cd /home/pi/alexa-avs-sample-app/samples/companionService && npm start
Run the AVS Java Client: cd /home/pi/alexa-avs-sample-app/samples/javaclient && mvn exec:exec
Run the wake word agent:
 Sensory: cd /home/pi/alexa-avs-sample-app/samples/wakeWordAgent/src && ./wakeWordAgent -e sensory
 KITT_AI: cd /home/pi/alexa-avs-sample-app/samples/wakeWordAgent/src && ./wakeWordAgent -e kitt_ai
 GPIO: PLEASE NOTE -- If using this option, run the wake word agent as sudo:
 cd /home/pi/alexa-avs-sample-app/samples/wakeWordAgent/src && sudo ./wakeWordAgent -e gpi o

real 76m2.884s
user 29m50.730s
sys 11m28.070s
pi@RPi0W:~/alexa-avs-sample-app \$

RPi Alexa

► Installation Completed



The screenshot shows a terminal window titled "pi@RPi0W: ~/alexa-...". The window displays the following command-line session:

```
pi@RPi0W:~/alexa-avc-sample-app$ ls
automated_install.sh  Installer_Licenses.txt  NOTICE.txt  RELEASE.txt
CONTRIBUTING.md        LICENSE.txt            README.md   samples
pi@RPi0W:~/alexa-avc-sample-app$ cd samples/
pi@RPi0W:~/alexa-avc-sample-app/samples$ ls
androidCompanionApp  companionService  iosCompanionApp  javaclient  wakeWordAgent
pi@RPi0W:~/alexa-avc-sample-app/samples$ 
pi@RPi0W:~/alexa-avc-sample-app/samples$ cd companionService/
pi@RPi0W:~/alexa-avc-sample-app/samples/companionService$ ls
app.js  authentication.js  bin  config.js  node_modules  package.json  template_config_js
pi@RPi0W:~/alexa-avc-sample-app/samples/companionService$ 
pi@RPi0W:~/alexa-avc-sample-app/samples/companionService$ npm start

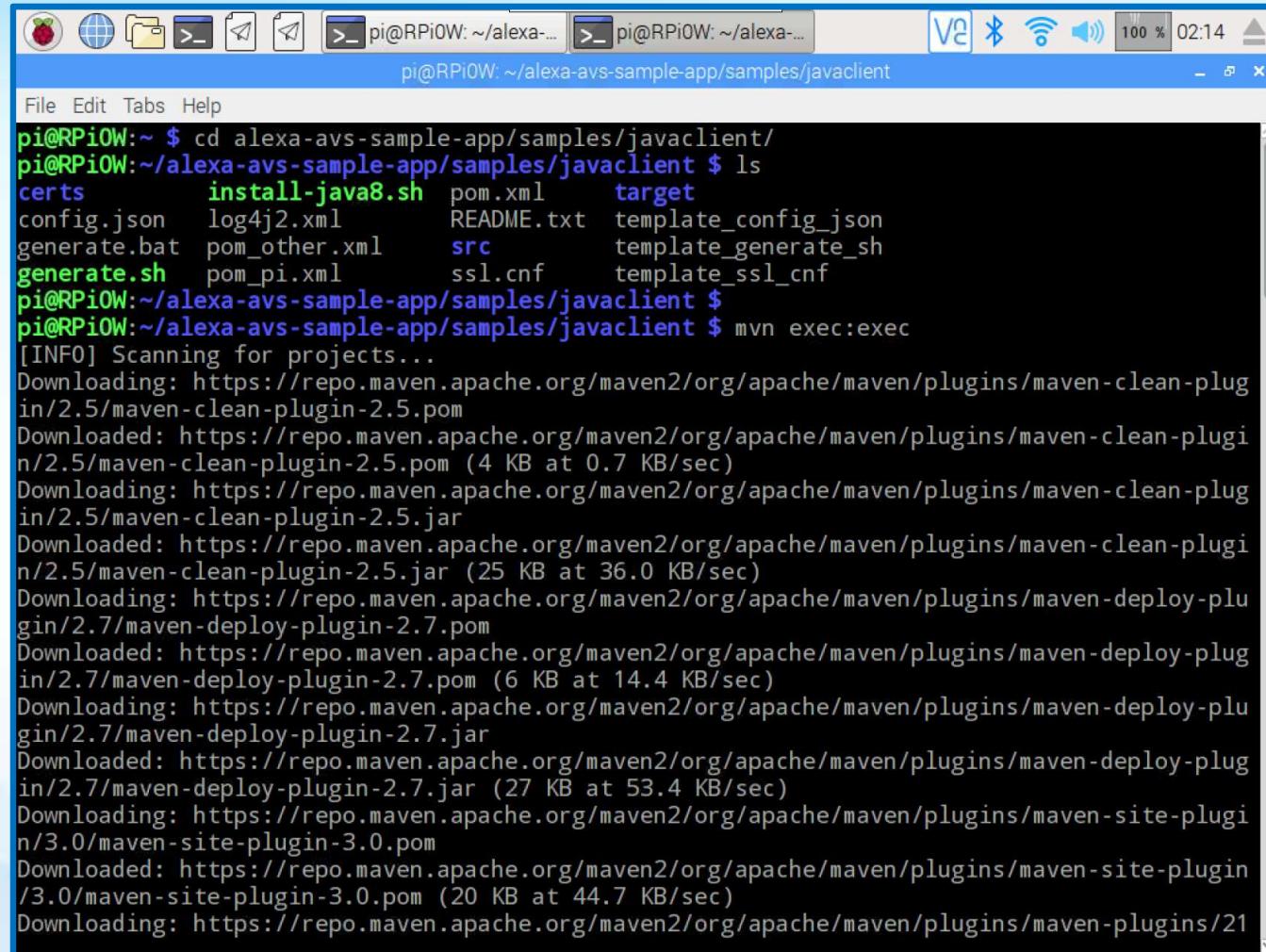
> alexa-voice-service-sample-companion-service@1.0.0 start /home/pi/alexa-avc-sample-app/samples/companionService
> node ./bin/www

This node service needs to be running to store token information memory and vend them for the AVS app.

Listening on port 3000
```

RPi Alexa

► Installation Completed

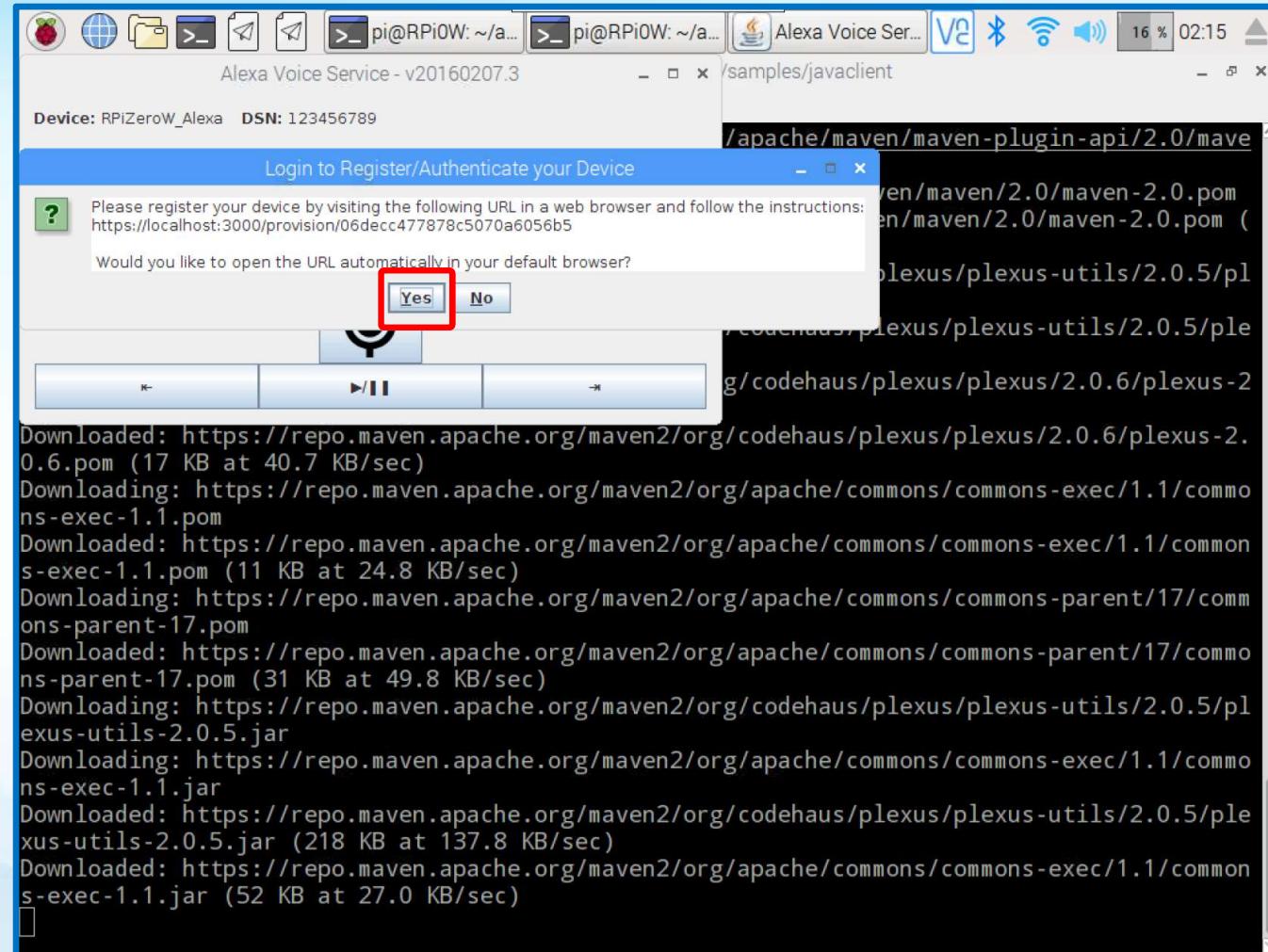


The screenshot shows a terminal window titled "pi@RPi0W: ~/alexa-avs-sample-app/samples/javaclient". The window contains the following command-line session:

```
pi@RPi0W:~ $ cd alexa-avs-sample-app/samples/javaclient/
pi@RPi0W:~/alexa-avs-sample-app/samples/javaclient $ ls
certs      install-java8.sh  pom.xml    target
config.json  log4j2.xml      README.txt  template_config_json
generate.bat pom_other.xml   src        template_generate_sh
generate.sh  pom_pi.xml     ssl.cnf    template_ssl_cnf
pi@RPi0W:~/alexa-avs-sample-app/samples/javaclient $
pi@RPi0W:~/alexa-avs-sample-app/samples/javaclient $ mvn exec:exec
[INFO] Scanning for projects...
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.pom
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.pom (4 KB at 0.7 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.jar (25 KB at 36.0 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.pom
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.pom (6 KB at 14.4 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.jar (27 KB at 53.4 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-site-plugin/3.0/maven-site-plugin-3.0.pom
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-site-plugin/3.0/maven-site-plugin-3.0.pom (20 KB at 44.7 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/21
```

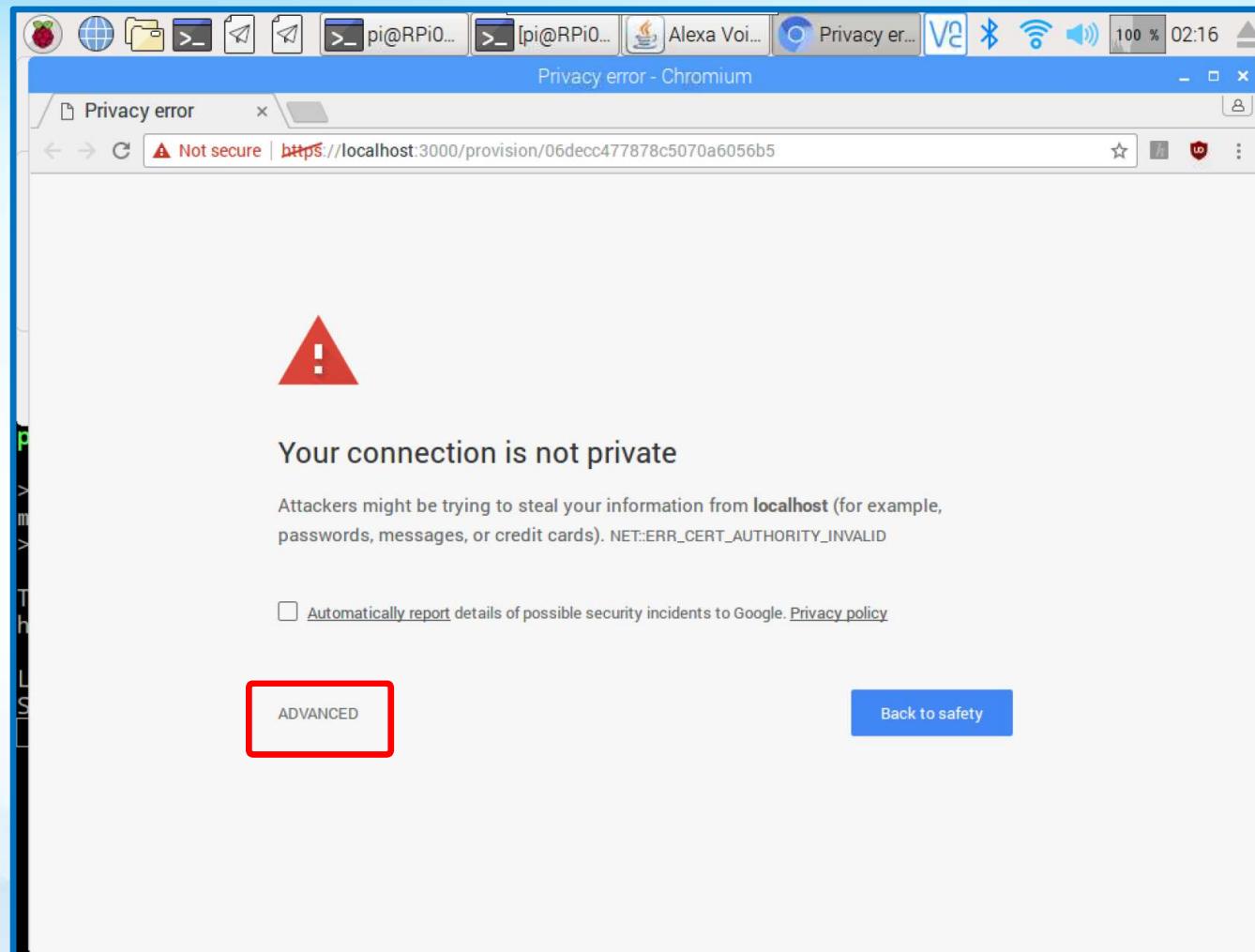
RPi Alexa

▶ Installation Completed



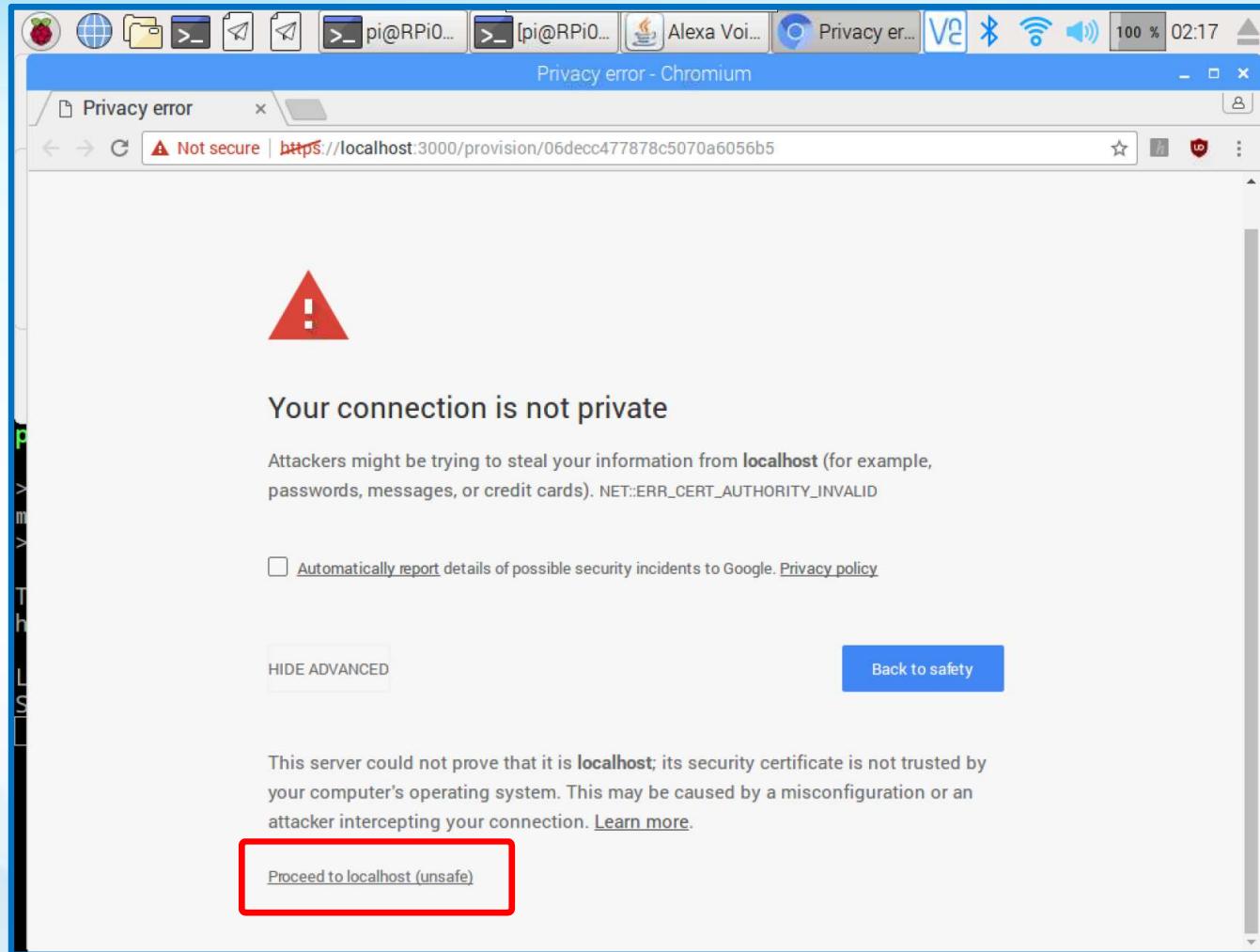
RPi Alexa

► Installation Completed



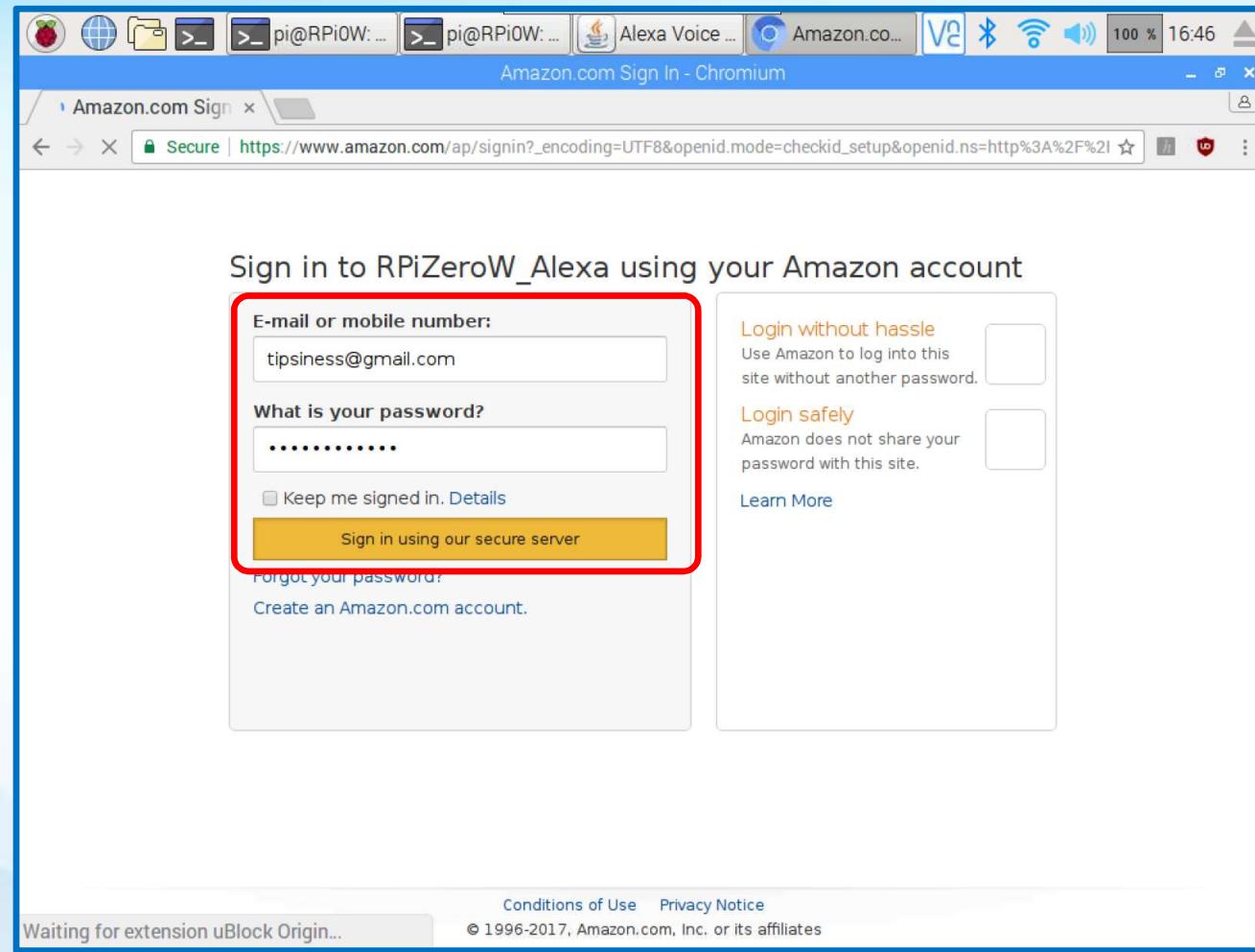
RPi Alexa

► Installation Completed



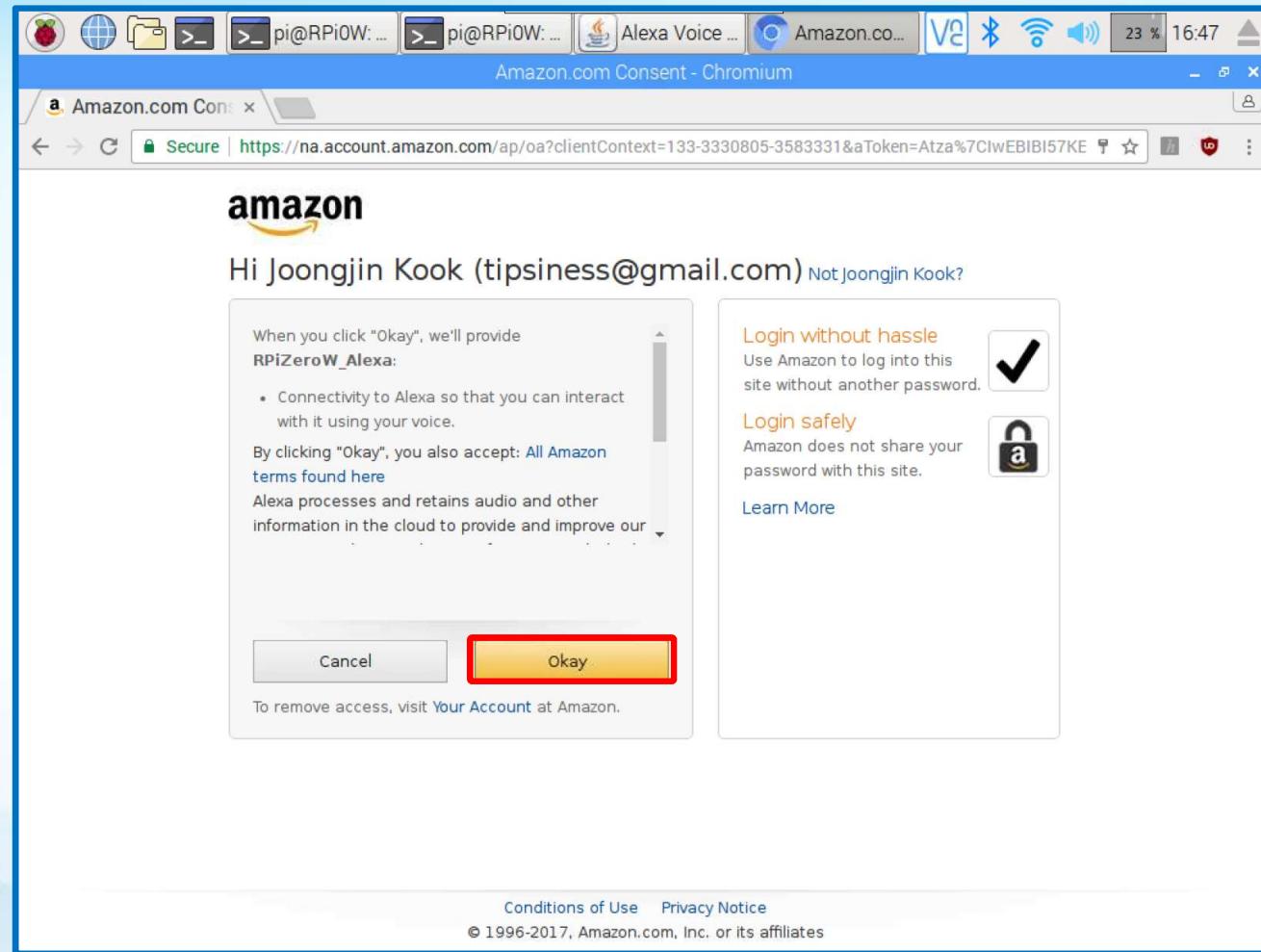
RPi Alexa

► Installation Completed



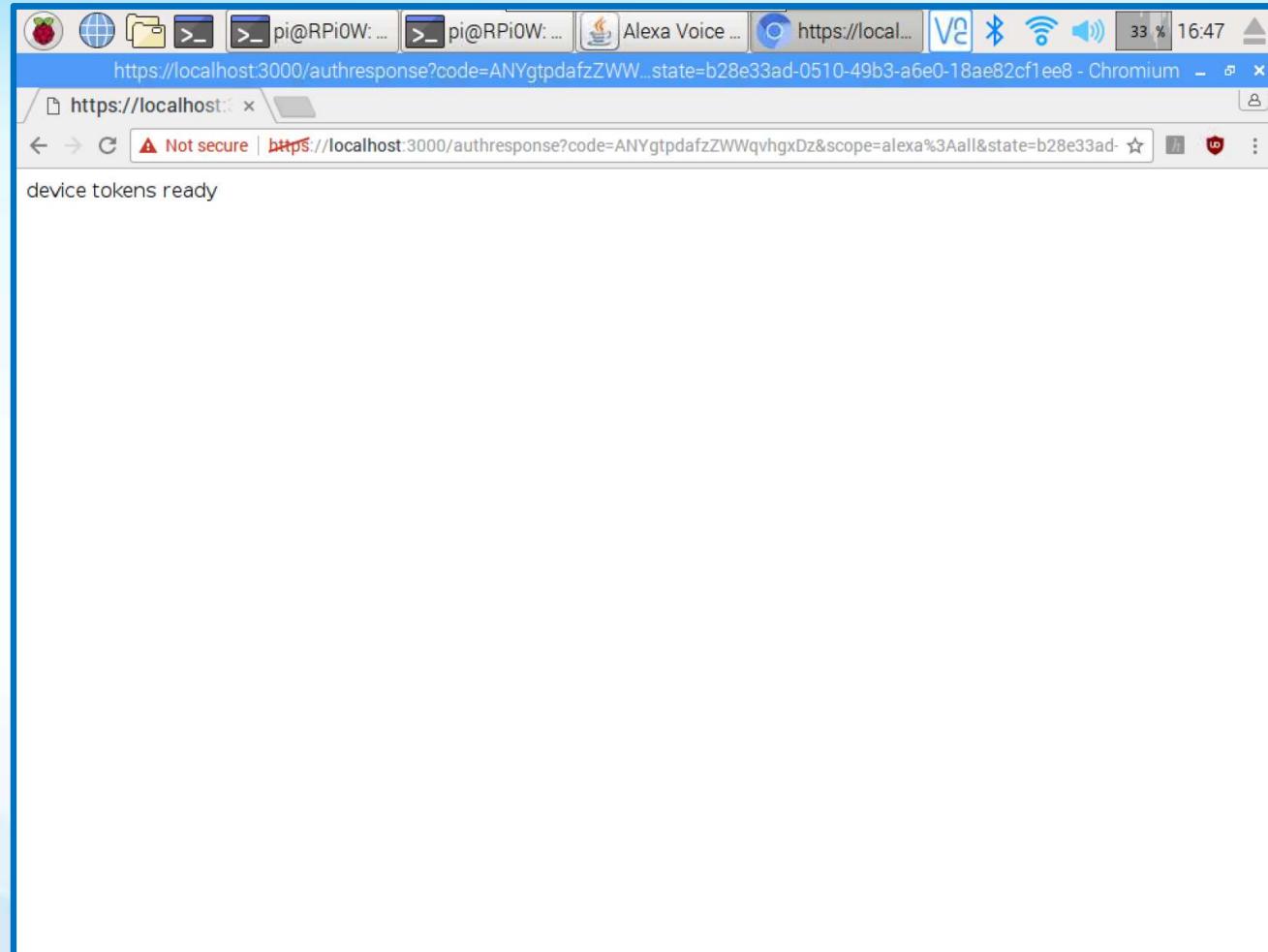
RPi Alexa

▶ Installation Completed



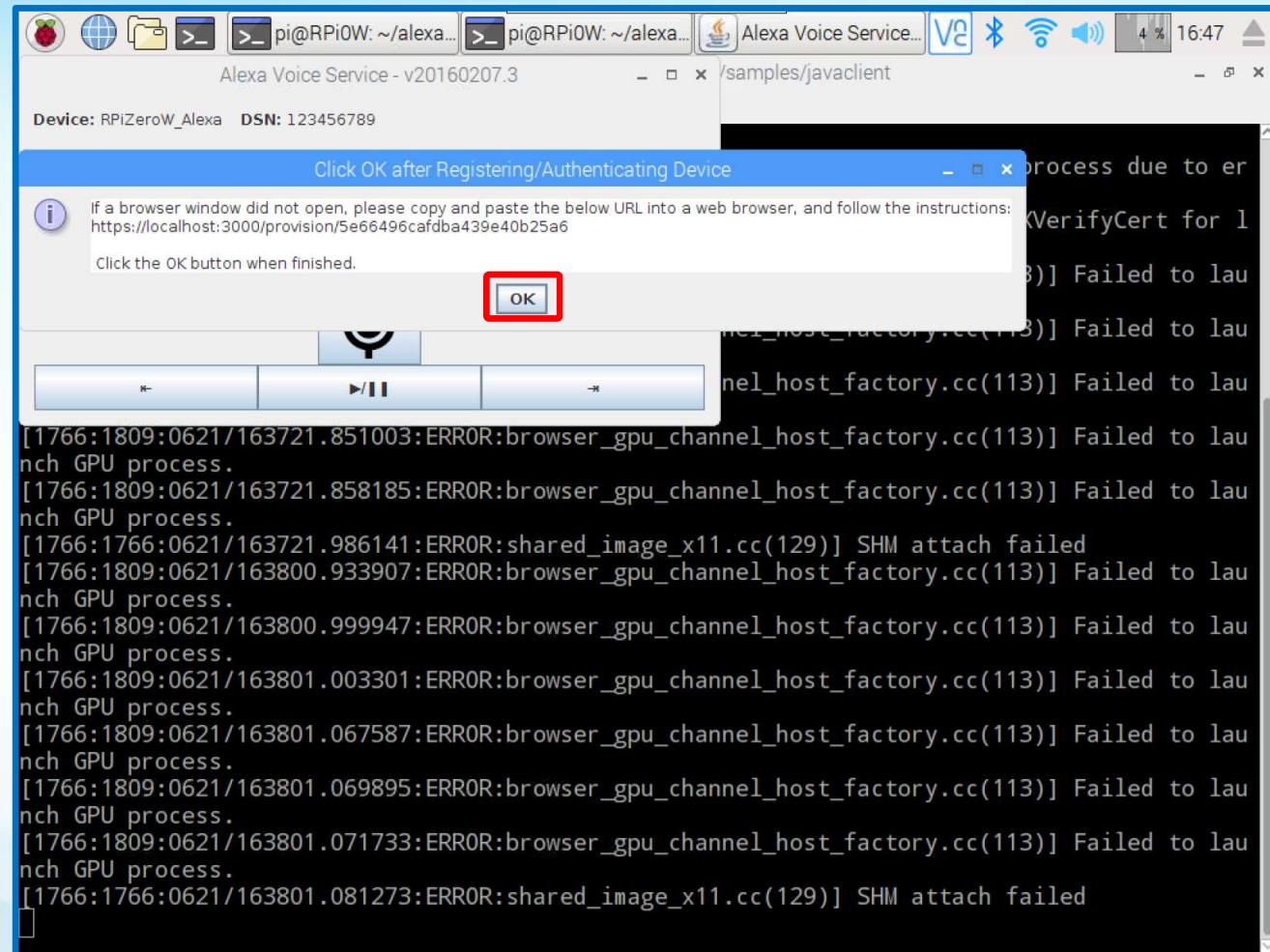
RPi Alexa

► Installation Completed



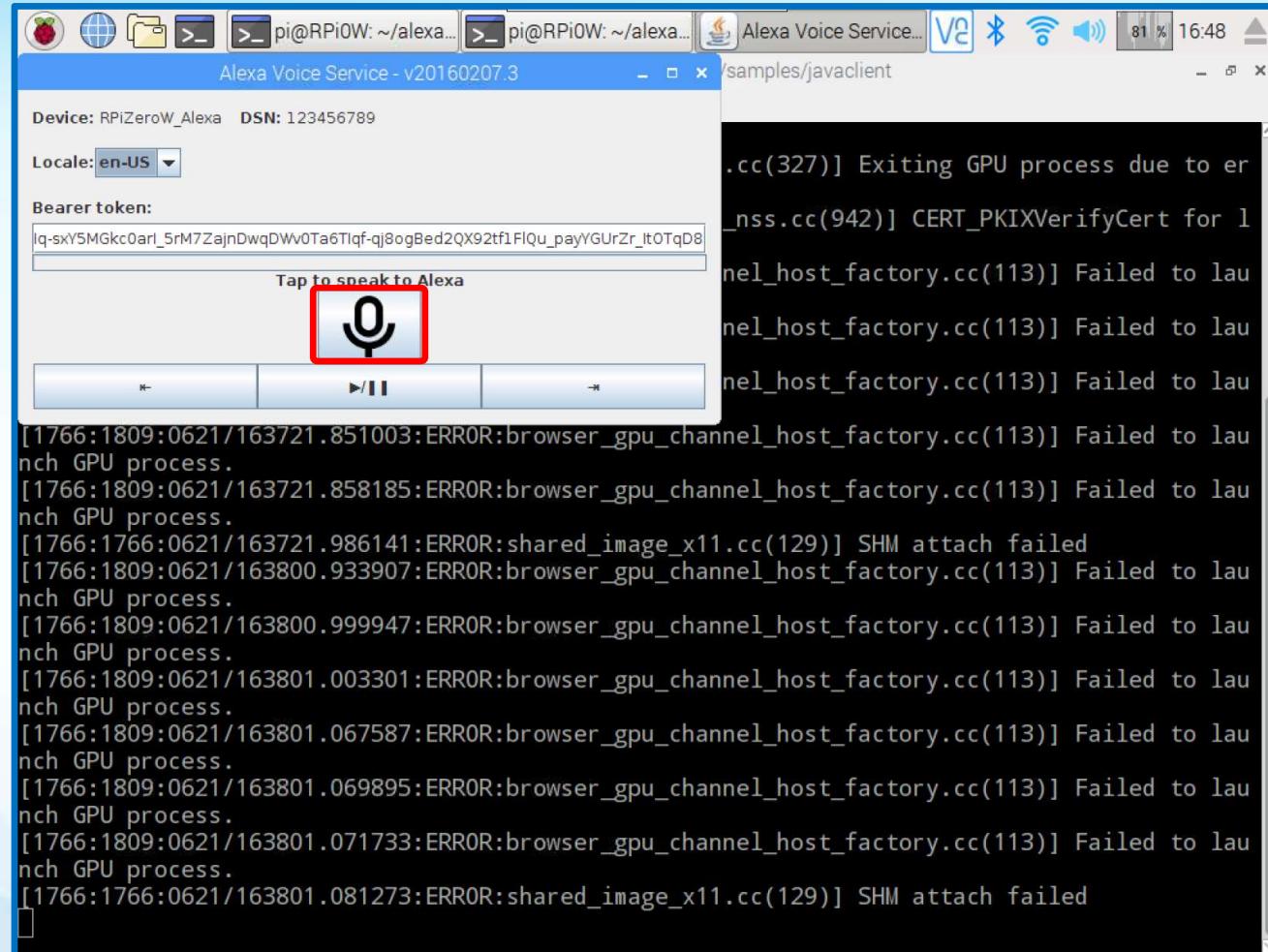
RPi Alexa

► Installation Completed



RPi Alexa

▶ Installation Completed



RPi Alexa

► Run

► Step1.

- \$ cd alexa-avs-sample-app/samples/companionService
- \$ npm start

```
pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/companionService $ ls
app.js          bin      node_modules  refresh_tokens
authentication.js config.js  package.json template_config_js
pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/companionService $ npm start
> alexa-voice-service-sample-companion-service@1.0.0 start /home/pi/Alexa/alexa-
avs-sample-app/samples/companionService
> node ./bin/www
```

This node service needs to be running to store token information memory and vend them for the AVS app.

Listening on port 3000

RPi Alexa

► Run

► Step1.

- \$ cd alexa-avs-sample-app/samples/companionService
- \$ npm start

```
pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/companionService $ ls  
app.js          bin        node_modules  refresh_tokens  
authentication.js config.js  package.json  template_config_js  
pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/companionService $ npm start  
  
> alexa-voice-service-sample-companion-service@1.0.0 start /home/pi/Alexa/alexa-  
avs-sample-app/samples/companionService  
> node ./bin/www
```

This node service needs to be running to store token information memory and vend them for the AVS app.

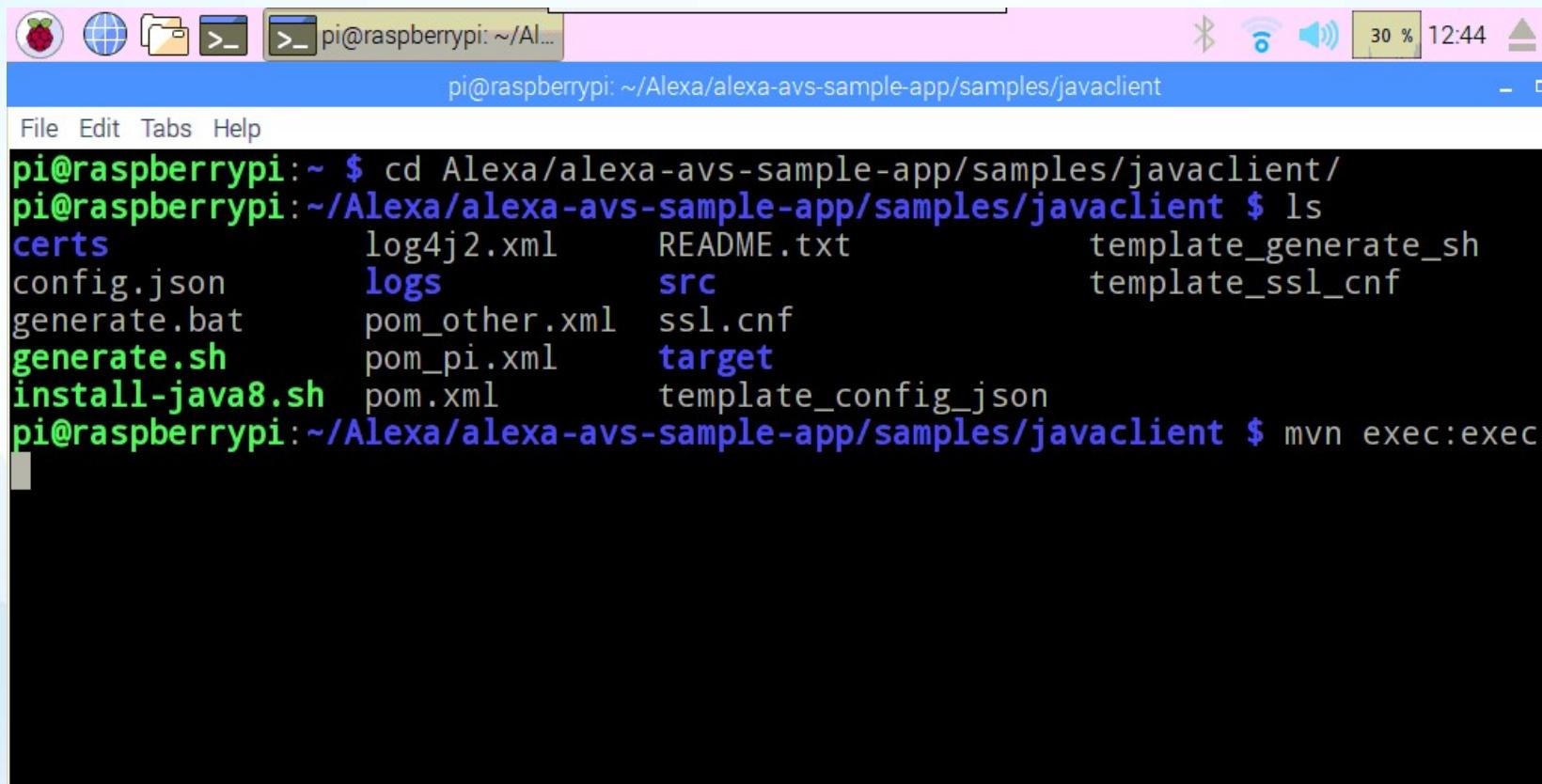
```
Listening on port 3000  
Successfully retrieved access token for session id: 07667818-72ea-45c3-87c1-c202  
952f2244
```

RPi Alexa

► Run

► Step2.

- \$ cd alexa-avs-sample-app/samples/javaclient
- \$ mvn exec:exec



The screenshot shows a terminal window on a Raspberry Pi. The title bar indicates the session is running as 'pi' on 'raspberrypi' at the root directory (~). The window contains the following command-line session:

```
pi@raspberrypi:~ $ cd Alexa/alexa-avs-sample-app/samples/javaclient/
pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/javaclient $ ls
certs          log4j2.xml      README.txt          template_generate_sh
config.json    logs           src                 template_ssl_cnf
generate.bat   pom_other.xml  ssl.cnf            target
generate.sh    pom_pi.xml    pom.xml           template_config_json
install-java8.sh pom.xml       pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/javaclient $ mvn exec:exec
```

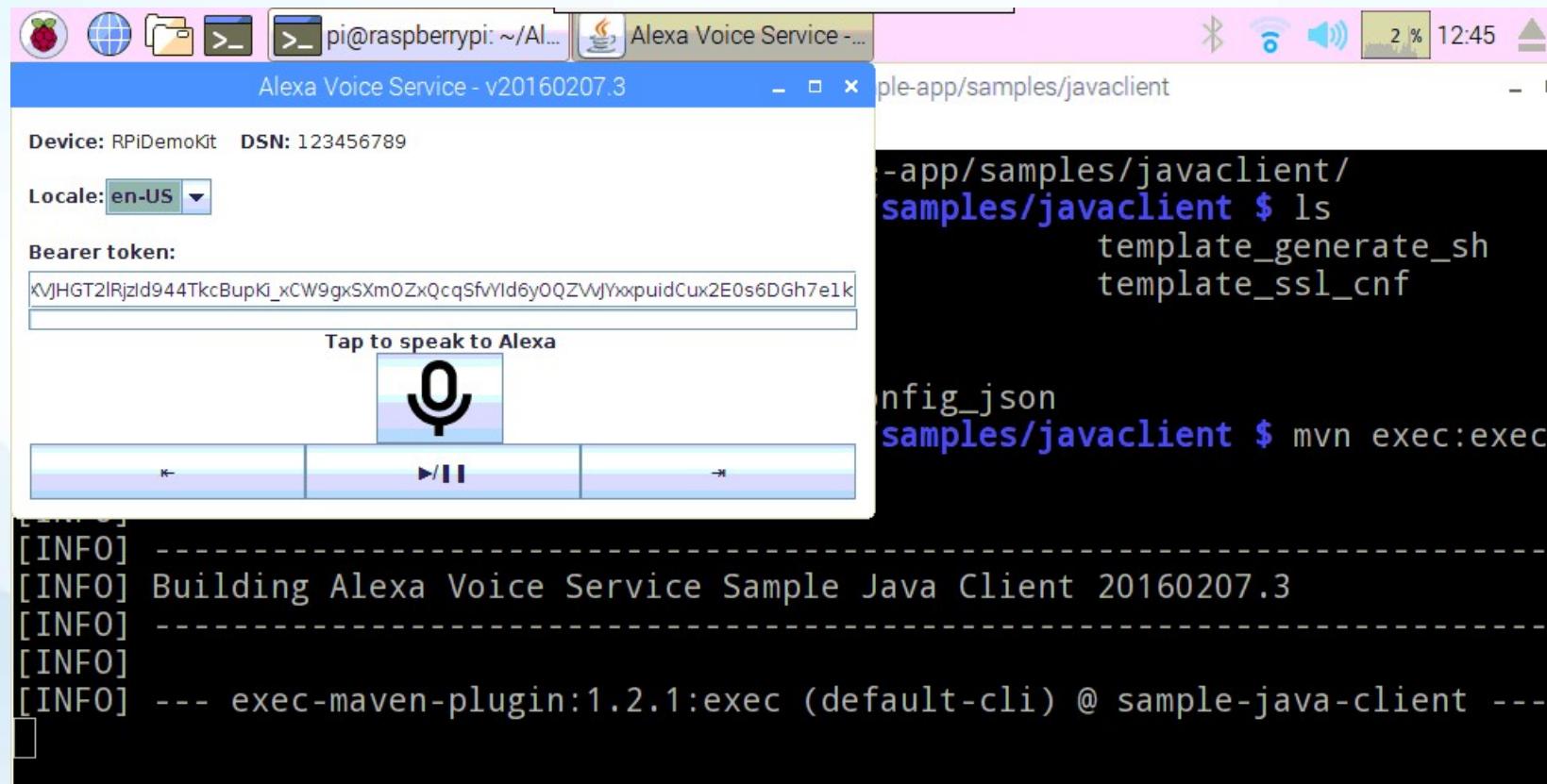
The terminal window also displays the system status bar at the top right, showing battery level (30%), time (12:44), and signal strength icons.

RPi Alexa

▶ Run

▶ Step2.

```
▶ $ cd alexa-avs-sample-app/samples/javaclient  
▶ $ mvn exec:exec
```



Amazon Alexa

► Use Wake Word Engine

- card # & device # of playback device, capture device

```
pi@raspberrypi:~ $ cat /proc/asound/modules
0 snd_bcm2835
1 snd_usb_audio
pi@raspberrypi:~ $ arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: Microphone [USB Microphone], device 0: USB Audio [USB Audio]
Subdevices: 0/1
Subdevice #0: subdevice #0
pi@raspberrypi:~ $
pi@raspberrypi:~ $ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: ALSA [bcm2835 ALSA], device 0: bcm2835 ALSA [bcm2835 ALSA]
Subdevices: 8/8
Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7
card 0: ALSA [bcm2835 ALSA], device 1: bcm2835 ALSA [bcm2835 IEC958/HDMI]
Subdevices: 1/1
Subdevice #0: subdevice #0
card 1: Microphone [USB Microphone], device 0: USB Audio [USB Audio]
Subdevices: 1/1
Subdevice #0: subdevice #0
pi@raspberrypi:~ $
```

Amazon Alexa

- ▶ Use Wake Word Engine
 - ▶ edit ~/.asoundrc

```
pcm.!default {  
    type asym  
    playback.pcm {  
        type plug  
        slave.pcm "hw:0,0"  
    }  
    capture.pcm {  
        type plug  
        slave.pcm "hw:1,0"  
    }  
}
```

Amazon Alexa

► Use Wake Word Engine

- ▶ Sensory's TrulyHandsFree and KITT.AI's Snowboy
- ▶ Sensory wake word engine
 - ▶ cd ~/alexa-avs-sample-app/samples/wakeWordAgent/src
 - ▶ ./wakeWordAgent -e sensory
- ▶ KITT.AI wake word engine
 - ▶ cd ~/alexa-avs-sample-app/samples/wakeWordAgent/src
 - ▶ ./wakeWordAgent -e kitt_ai

Amazon Alexa

- ▶ Use Wake Word Engine
 - ▶ Sensory wake word engine

```
pi@raspberrypi:~/Alexa/alexa-avs-sample-app/samples/wakeWordAgent/src $ ./wakeWordAgent -e sensory
INFO:main: Starting Wake Word Agent
INFO:WakeWordAgent: State set to IDLE(2)
INFO:Initializing Sensory library | library name: TrulyHandsfree | library version: 5.0.0-beta.10.2 | model file: ../ext/resources/spot-alexa-rpi.snsr
WARNING:Library expires on: License expires on 28 Sep 2017 00:00:00 GMT
INFO:SensoryWakeWordEngine: mainLoop thread started
INFO:WakeWordIPCSocket::mainLoop thread started
INFO:WakeWordIPCSocket: init socket on port:5123
INFO:WakeWordAgent: thread started
INFO:====> Connected to AVS client <===
INFO:====> WakeWordAgent: wake word detected <===
INFO:WakeWordAgent: State set to WAKE_WORD_DETECTED(3)
INFO:WakeWordAgent: State set to SENT_WAKE_WORD_DETECTED(4)
INFO:WakeWordAgent: IPC Command received:3
INFO:WakeWordAgent: State set to WAKE_WORD_PAUSE_REQUESTED(5)
INFO:SensoryWakeWordEngine: handling pause
INFO: *** THREAD JOINING: Sensory ***
INFO:SensoryWakeWordEngine: mainLoop thread ended
INFO:WakeWordAgent: State set to WAKE_WORD_PAUSED(6)
INFO:WakeWordAgent: IPC Command received:4
INFO:WakeWordAgent: State set to WAKE_WORD_RESUME_REQUESTED(7)
INFO:SensoryWakeWordEngine: handling resume
Say, "Alexa!"
```

Amazon Alexa

► ARTIK Cloud, works with Alexa

The screenshot shows the ARTIK Cloud website's "Works with..." page for the Amazon Echo. The top navigation bar includes links for "Works with...", "Pricing", "Blog", "Contact Us", "Developer", "My ARTIK Cloud", and a search icon. Below the navigation, a breadcrumb trail reads "Works with... > Amazon Echo". On the left, there's a large image of an Amazon Echo device with its logo. Below it, the text "Amazon Echo" is displayed, followed by a description: "Hands-free speaker that fills any room with immersive 360° audio". Two buttons are present: "Personal Assistant" and "Learn more". At the bottom of this section, the text "Also in Personal Assistant" is visible. On the right, a descriptive text block states: "If you have an Amazon Echo or use Alexa for voice recognition you will love our integration." Below this, a heading "How to connect it to ARTIK Cloud" is followed by instructions: "Install the ARTIK Cloud Skill and control any device connected to ARTIK Cloud. Go to the Alexa website and say to Alexa "Connect ARTIK Cloud". Wait for Alexa to say "Welcome to Samsung ARTIK Cloud, I am currently retrieving your devices" and shortly after a list of the devices that were recognized." Further down, it says: "You can now instruct Alexa to interact with those devices. A new device named **Amazon Alexa proxy** will also be automatically created in your My ARTIK Cloud profile. Now ask Alexa to do something, say "Alexa Ask ARTIK to..." and complete the sentence with a command." A note at the bottom states: "Please note - the ARTIK Cloud Skill can only be added on the Alexa website."

Amazon Alexa

► Getting Started with Alexa

Getting Started with Alexa

Welcome, Joongjin Kook!

Amazon processes and retains audio, interactions, and other data in the cloud to provide and improve our services. Alexa allows purchasing by voice using your default payment and shipping settings. You can require a speakable confirmation code, turn off voice purchasing, and see product and order details in your Alexa app or on alexa.amazon.com. Alexa also allows you to call and message your Alexa contacts. [Learn more.](#)

By clicking "Continue", you agree to [all the terms found here.](#)

[Continue](#)

[Sign out](#)

Amazon Alexa

Alexa Home

The screenshot displays the Amazon Alexa Home interface. At the top, a dark header bar contains the word "Home". Below it, a light-colored sidebar on the left lists various options: "Now Playing", "Music & Books", "Lists", "Reminders & Alarms", "Skills", "Smart Home", "Things to Try", "Settings", "Help & Feedback", and "Not Joongjin? Sign out". The main content area on the right is titled "Home" and features a section titled "Things to Try" with the sub-instruction "*"Alexa, what's on my calendar?"*". It also shows a message about Apple's iCloud calendar being available. Below this, a large "Welcome!" message is displayed, followed by a paragraph from Amazon stating they are excited to bring Alexa to the RPi-Alexa Demo Kit. A "Customize Alexa" button with a "Learn more" link is present, along with a "Remove card" button and a "Less" link. At the bottom of the main content area, there is a "More" link.

Amazon Alexa

► Rules

The screenshot shows the Amazon Alexa mobile application interface. On the left is a vertical navigation bar with the following options:

- Home
- Now Playing
- Music & Books
- Lists
- Reminders & Alarms
- Skills
- Smart Home
- Things to Try
- Settings** (this option is highlighted with a red rectangle)
- Help & Feedback
- Not Joongjin? Sign out

The main content area is titled "Settings" at the top. It contains two sections: "Devices" and "Accounts".

Devices section:

- Joongjin's RPi-Alexa Demo Kit (status: Offline)
- Set up a new device

Accounts section:

- Music & Media
- Flash Briefing
- Sports Update
- Traffic
- Calendar
- Lists
- Voice Training
- Voice Purchasing
- Household Profile

A red box highlights the "Devices" section, and the text "RPi Alexa" is overlaid in red on the right side of the "Devices" section.

Amazon Alexa

► Rules

The screenshot shows the Alexa mobile app interface. On the left is a vertical navigation bar with the following options: Home, Now Playing, Music & Books, Lists, Reminders & Alarms, Skills, Smart Home, Things to Try, **Settings**, Help & Feedback, and Not Joongjin? Sign out. The "Settings" option is highlighted with a red box. The main content area is titled "Settings" and contains several sections:

- Do Not Disturb**: A switch labeled "Off" is shown.
- General**: This section is also highlighted with a red box. It includes:
 - Device name**: Set to "Joongjin's RPi-Alexa Demo Kit" with an "Edit" link.
 - Device location**: A note says "This location will be used for weather and other local features." with a link "Add a street address for better local information". The location is listed as "98109" with an "Edit" link.
 - Device time zone**: Set to "United States" (Pacific Daylight Time, Los Angeles).
- Measurement Units**:
 - Temperature Units**: A switch labeled "Off". A note says "Use metric measurements for temperature units."
 - Distance Units**: A switch labeled "Off". A note says "Use metric measurements for distance units."

Amazon Alexa

► Rules

The screenshot shows the 'Now Playing' screen of the Amazon Alexa mobile application. On the left is a vertical navigation menu with options: Home, Now Playing (which is selected), Music & Books, Lists, Reminders & Alarms, Skills, Smart Home, Things to Try, Settings, Help & Feedback, and Not Joongjin? Sign out. At the top center, it says 'Now Playing'. To the right of the menu, there are tabs for Queue (which is active) and History. A red rectangular box highlights a message box in the center. The message box contains a Wi-Fi signal icon with a crossed-out 'X', followed by the text 'Your Alexa device is not connected'. Below that, it says 'To play music and media, your Alexa device must be online. Go to Wi-Fi setup.' To the right of the message box, the text 'RPi Alexa Disabled' is overlaid in red.

Amazon Alexa

► Rules

The screenshot shows the Amazon Alexa mobile application interface. On the left is a vertical navigation bar with the following options: Home, Now Playing (which is selected and highlighted in grey), Music & Books, Lists, Reminders & Alarms, Skills, Smart Home, Things to Try, Settings, Help & Feedback, and Not Joongjin? Sign out. The main content area is titled "Now Playing". It displays a message: "Amazon Alexa can play any music, radio station, or show you like - just ask. Here are some examples:" followed by a list of voice command examples. A red rectangular box highlights this message and example list. At the bottom right of the main content area, the text "RPi Alexa Enabled" is displayed in red. The top right of the screen shows three tabs: Queue (which is underlined in orange) and History.

Now Playing

Queue History

Amazon Alexa can play any music, radio station, or show you like - just ask.
Here are some examples:

"Alexa, play music."
"Alexa, play Jazz music from Prime."
"Alexa, play music by Bruno Mars."
"Alexa, read my Audible book."
"Alexa, play All Things Considered from TuneIn."

RPi Alexa Enabled

Amazon Alexa

▶ Alexa Skills for ARTIK Cloud

The screenshot shows a web browser window titled "Amazon Alexa" with the URL "alexa.amazon.com/spa/index.html#skills/search/artik/?&ref-suffix=sb_gw". The left sidebar has a red box around the "Skills" link. The main content area is titled "Search" with the search term "artik" in the input field. A red box highlights the first result, which is "Samsung ARTIK Cloud" by Samsung Strategy and Innovation Center, with a 4-star rating and the phrase "Alexa, start ARTIK". The second result, also highlighted with a red box, is "Samsung ARTIK Cloud Optimized for Smart Home" by Samsung Strategy and Innovation Center, with a 1-star rating and the phrase "Alexa, turn on bedroom lights".

Home

Now Playing

Music & Books

Lists

Reminders & Alarms

Skills

Smart Home

Things to Try

Settings

Help & Feedback

Not Joongjin? Sign out

Search

artik

2 RESULTS

Samsung ARTIK Cloud

Samsung Strategy and Innovation Center

★★★★★ 4

"Alexa, start ARTIK"

Samsung ARTIK Cloud Optimized for Smart Home

Samsung Strategy and Innovation Center

★★★★★ 1

"Alexa, turn on bedroom lights"

Amazon Alexa

▶ Alexa Skills for ARTIK Cloud

The screenshot shows a web browser window titled "Amazon Alexa" with the URL "alexa.amazon.com/spa/index.html#skills/dp/B01E8NGW60". The main content is titled "Samsung ARTIK Cloud". On the left is a sidebar with links: Home, Now Playing, Music & Books, Lists, Reminders & Alarms, Skills (which is selected), Smart Home, Things to Try, Settings, Help & Feedback, and Not Joongjin? Sign out. The main area displays the skill information for "Samsung ARTIK Cloud" by "Samsung Strategy and Innovation Center". It has a 4-star rating. A large blue "ENABLE" button is highlighted with a red border. Below it, a message says "Account linking required". Under "TRY SAYING", there are three examples: "Alexa, start ARTIK", "turn on the living room light", and "open the front door". Under "ABOUT THIS SKILL", it says: "Samsung ARTIK Cloud is an agnostic IoT platform: <https://www.artik.cloud>". It describes the skill as letting you command any device connected to the ARTIK Cloud platform. It also explains how to tell ARTIK what action to perform on which device, giving the example of turning on a living room light. Finally, it provides instructions to get started, starting with creating an account on the ARTIK Cloud user portal.

Home

Now Playing

Music & Books

Lists

Reminders & Alarms

Skills

Smart Home

Things to Try

Settings

Help & Feedback

Not Joongjin? Sign out

Samsung ARTIK Cloud

Samsung Strategy and Innovation Center

4

ENABLE

Account linking required

TRY SAYING

"Alexa, start ARTIK"

"turn on the living room light"

"open the front door"

ABOUT THIS SKILL

Samsung ARTIK Cloud is an agnostic IoT platform: <https://www.artik.cloud>

The Samsung ARTIK Cloud Alexa Skill lets you command by voice any device connected to the ARTIK Cloud platform.

You can tell ARTIK which action you want to perform on which device. For example, to command your living room light, you may tell ARTIK to "turn on the living room light" or "turn off the living room light".

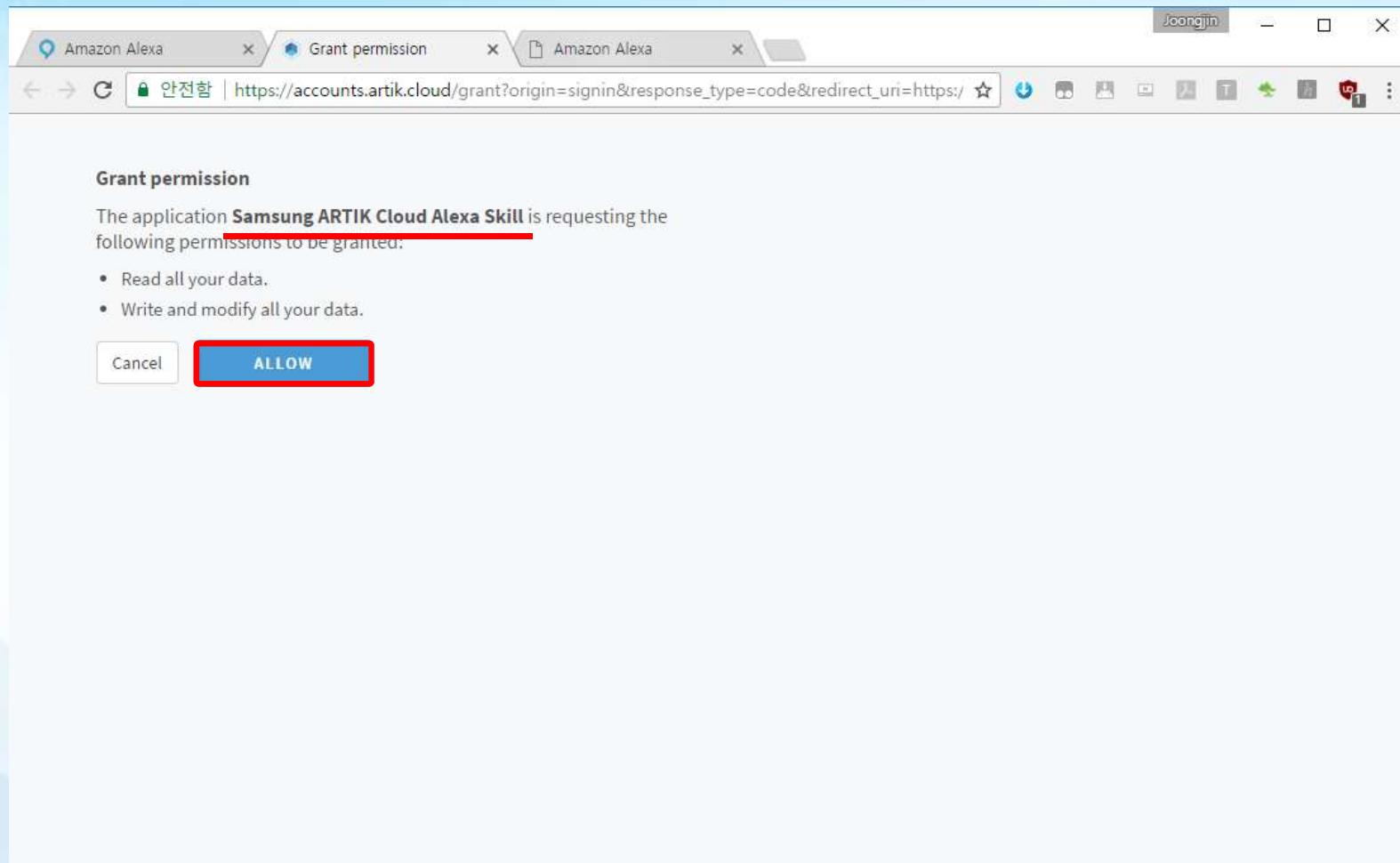
To get started:

1. Create an account on the ARTIK Cloud user portal: <https://www.artik.cloud...>

[See More](#)

Amazon Alexa

▶ Alexa Skills for ARTIK Cloud



Amazon Alexa

▶ Alexa Skills for ARTIK Cloud

The screenshot shows a web browser window titled "Amazon Alexa" with the URL "alexa.amazon.com/spa/index.html#skills/dp/B01HYC3KRQ". The main content is titled "Samsung ARTIK Cloud Optimized for Smart Home". On the left is a sidebar with links: Home, Now Playing, Music & Books, Lists, Reminders & Alarms, Skills (which is selected), Smart Home, Things to Try, Settings, Help & Feedback, and Not Joongjin? Sign out. The main area features the skill logo, name, developer information (Samsung Strategy and Innovation Center, 1 star rating), and an "ENABLE" button which is highlighted with a red box. Below the button, it says "Account linking required". Under "TRY SAYING", there are three examples: "Alexa, turn on bedroom lights", "Alexa, turn off bedroom lights", and "Alexa, set bedroom lights to 20%". In the "ABOUT THIS SKILL" section, it says: "The Samsung ARTIK Cloud Smart Home Skill enables you to use your voice to interact with devices connected to ARTIK Cloud." It also provides a link to "https://www.artik.cloud" and instructions for getting started.

Samsung ARTIK Cloud Optimized for Smart Home

Samsung ARTIK Cloud
Optimized for Smart Home

Samsung Strategy and Innovation Center

ENABLE

Account linking required

TRY SAYING

"Alexa, turn on bedroom lights"

"Alexa, turn off bedroom lights"

"Alexa, set bedroom lights to 20%"

ABOUT THIS SKILL

The Samsung ARTIK Cloud Smart Home Skill enables you to use your voice to interact with devices connected to ARTIK Cloud.

<https://www.artik.cloud>

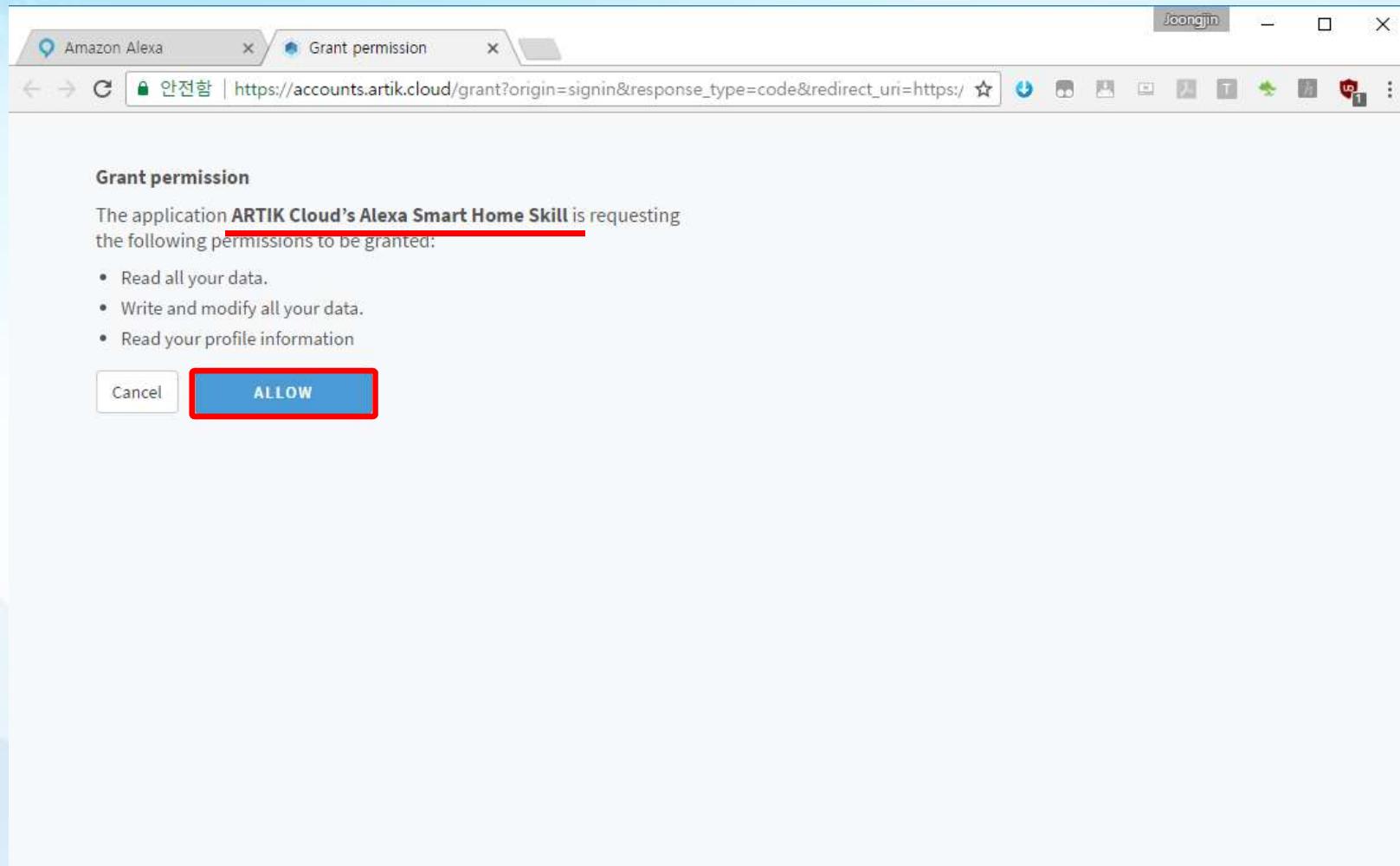
For example, to control lights in your living room light, say "Alexa, turn on the living room lights"

To get started:

- 1.Create an account on ARTIK Cloud (FREE to start): <https://www.artik.cloud>
2. Add your devices (follow the instructions on the website)
3. In the Alexa app, go to "Smart Home", search for "Samsung ARTIK Cloud Smart Home Skill" and link your ARTIK Cloud account....

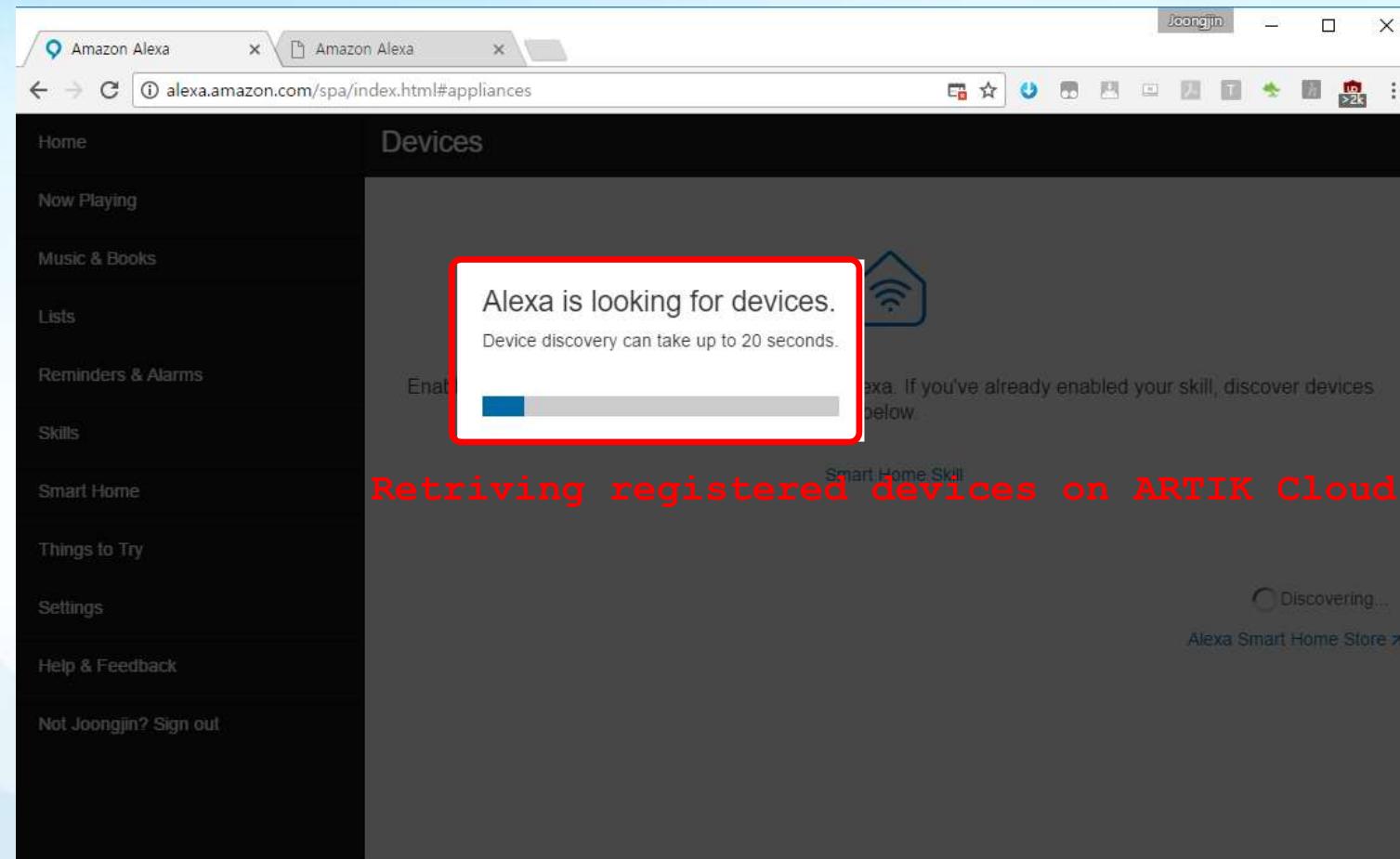
Amazon Alexa

▶ Alexa Skills for ARTIK Cloud



Amazon Alexa

▶ Alexa Skills for ARTIK Cloud



Amazon Alexa

Alexa Skills for ARTIK Cloud

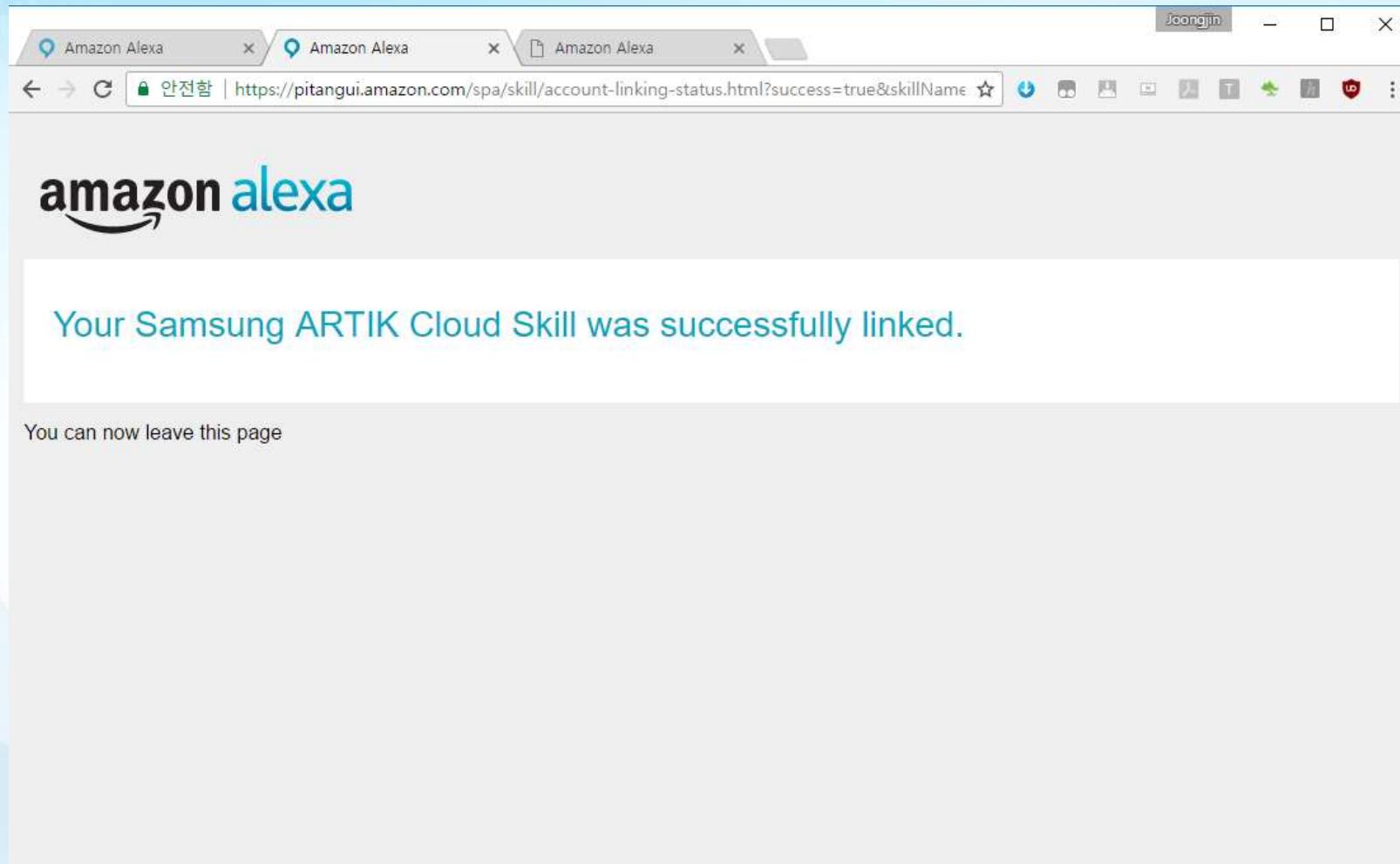
The screenshot shows the Alexa Devices page with a sidebar on the left and a main content area on the right. The sidebar includes links for Home, Now Playing, Music & Books, Lists, Reminders & Alarms, Skills, Smart Home, Things to Try, Settings, Help & Feedback, and a sign-out link. The main content area is titled 'Devices' and displays a list of registered devices. A red box highlights the first four items in the list:

Device Name	Type	Action
red lamp	Samsung ARTIK Cloud - ARTIK Smart Parking LED	Forget
green lamp	Samsung ARTIK Cloud - ARTIK Smart Parking LED	Forget
blue lamp	Samsung ARTIK Cloud - ARTIK Smart Parking LED	Forget
My ARTIK LED	Samsung ARTIK Cloud - ARTIK Smart Parking LED	Forget
AirCleaner_gtp	Samsung ARTIK Cloud - AirCleaner	Forget
ARTIK053_AirCleaner	Samsung ARTIK Cloud - AirCleaner	Forget
Kooks ARTIK Smart Parking LED	Samsung ARTIK Cloud - ARTIK Smart Parking LED	Forget
Kooks ARTIK 053	Samsung ARTIK Cloud - ARTIK 053	Forget

Retrieving registered devices on ARTIK Cloud

Amazon Alexa

▶ Alexa Skills for ARTIK Cloud



Amazon Alexa

▶ Say to Alexa

- ▶ "Start ARTIK"
- ▶ "red/green/blue lamp setOn/setOff"

The screenshot shows the Amazon Alexa Actions history interface. On the left, there's a sidebar with sections for "SHOW ACTIONS SENT EARLIER THAN" (a calendar for June 2017 with the 19th selected), "SHOW ACTIONS SENT TO DEVICE" (with "All" selected), and "SHOW ACTIONS BY NAME" (with "blue lamp" selected). The main area has tabs for "FILTERS", "MESSAGES" (selected), and "ACTIONS". Below is a table of actions:

DESTINATION DEVICE	SENT AT	SOURCE DEVICE	ACTION NAME	ACTION PARAMETERS
red lamp ARTIK Smart Parking LED 0681c153aa4d4d8c9...	Jun 19 2017 20:32:23.439	red lamp 0681c153aa4d4d8c9...	setOff	None
blue lamp ARTIK Smart Parking LED f5e4b35eb7d8430695..	Jun 19 2017 20:32:11.235	blue lamp f5e4b35eb7d8430695..	setOff	None
green lamp ARTIK Smart Parking LED 80345921edf0436eb9..	Jun 19 2017 20:30:28.433	green lamp 80345921edf0436eb9..	setOff	None
red lamp ARTIK Smart Parking LED 0681c153aa4d4d8c9...	Jun 19 2017 20:29:34.646	red lamp 0681c153aa4d4d8c9...	setOn	None
blue lamp ARTIK Smart Parking LED f5e4b35eb7d8430695..	Jun 19 2017 20:28:31.859	blue lamp f5e4b35eb7d8430695..	setOn	None
green lamp ARTIK Smart Parking LED 80345921edf0436eb9..	Jun 19 2017 20:27:56.017	green lamp 80345921edf0436eb9..	setOn	None
blue lamp	Jun 19 2017	blue lamp	setOff	None