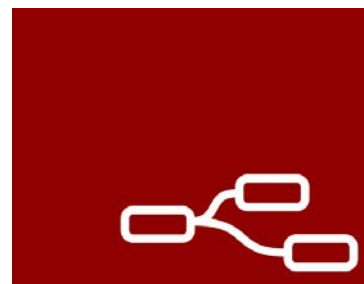


抜粋版

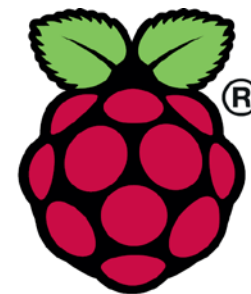
AWS IoTを使用したサーモセンサ開発キット ～食品輸送、人検知、機械故障等の温度管理まで幅広く応用～ 実践編



AWS IoT Core



Node-RED



Raspberry Pi

スペクトラム・テクノロジー株式会社

<https://spectrum-tech.co.jp>

sales@spectrum-tech.co.jp

サーモセンサ開発キット 目次

Pi運用マニュアル

1. RaspberryPiについて
2. Linux基本コマンド
3. 基本操作
4. 日常運用(ウイルススキャン、更新)

ページ

[3](#)

[4](#)

[5](#)

[6](#)

抜粋版のため本文
と一致しません

AWS IoT開発

- ① メニュー
- ② AWS設定
- ③ AWS IoT設定
- ④ サーモセンサ情報収集、アラーム送付
 - 全体構成
 - サーモセンサ概要
 - サーモセンサーRaspberryPi接続
 - サーモセンサ単体試験
 - サーモセンサ画像試験
 - Mqttプログラム設定
 - AWS IoT折り返し試験
 - リアルタイム表示(グラフ)
 - IoTイベント設定
 - IoTイベント試験
 - 遠隔リアルタイムモニタ
 - CSV取り出し
 - CSV画像再生

ページ

[8](#)

[9](#)

[16](#)

[29](#)

[30](#)

[31](#)

[32](#)

[33](#)

[34](#)

[36](#)

[37](#)

[39](#)

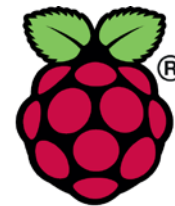
[47](#)

[50](#)

[51](#)

[53](#)





Pi運用マニュアル

1. Raspberry Piについて

既に全世界で1000万台以上販売された手のひらサイズのコンピュータです。
LinuxベースのRasbianOSで動作しております。

2. Linux基本コマンド

① システム関係

- 起動: 電源を入れると自動で起動します。
- 再起動: # reboot
又は、アプリケーション>ログアウト>再起動; 左上のメニューから
- 終了: # shutdown
又は、アプリケーション>ログアウト>シャットダウン; 左上のメニューから
- ログアウト # logout
又は、アプリケーション>ログアウト>ログアウト; 左上のメニューから
- **日本語／英語の入力切替**: キーボードのctl+jを同時に押します。又は右上のアイコン(右から7個目)からプルダウンで選択



Raspberry Pi

Pi運用マニュアル

2. Linux基本コマンド

② ディレクトリ操作、コピー、移動、削除

root@:~# **cd** /root/Documents ディレクトリの切り替え
 root@:/root/Documents# **ls** ファイルとディレクトリの表示(表示したら操作したいファイルを右クリックでコピーして操作します)
 root@:~# **cp** ファイル名 ディレクトリ 配下のディレクトリのファイルを別のディレクトリへコピー
 root@:~# **mv** ファイル名 ディレクトリ 配下のディレクトリのファイルを別のディレクトリへ移動
 root@:~# **rm** ファイル名 ファイルの削除
 便利な機能 **rm -help** コマンドのオプションが分からない場合は、ヘルプで問い合わせる。すべてのコマンド共通(マイナスを2個とhelp)

③ ユーザ権限、プロセス他

root@:~ \$ **su -** スーパーユーザ(root)に切り替え、パスワードを入力
 root@:~# **ps a** 現状の動いているプロセスを表示
 root@:~# **kill** 特定のプロセスを強制終了
 root@:~# **apt-get install pkg** パッケージのインストールなどに使用
 root@:~# **date** 日付、時間の設定を行います。
 root@:~# **leafpad** /etc/network/interfaces インタフェースに記述している内容を変更します。Viよりも使いやすいです。

④ モジュール、usb、メモリ、HDDなどの表示

root@:~# **lsmod** linuxのモジュールリスト表示
 root@:~# **lsusb** usbのデバイス表示
 root@:~# **free -mt** メモリ使用状態表示
 root@:~# **df -h** HDD(マイクロSD)の使用状態表示



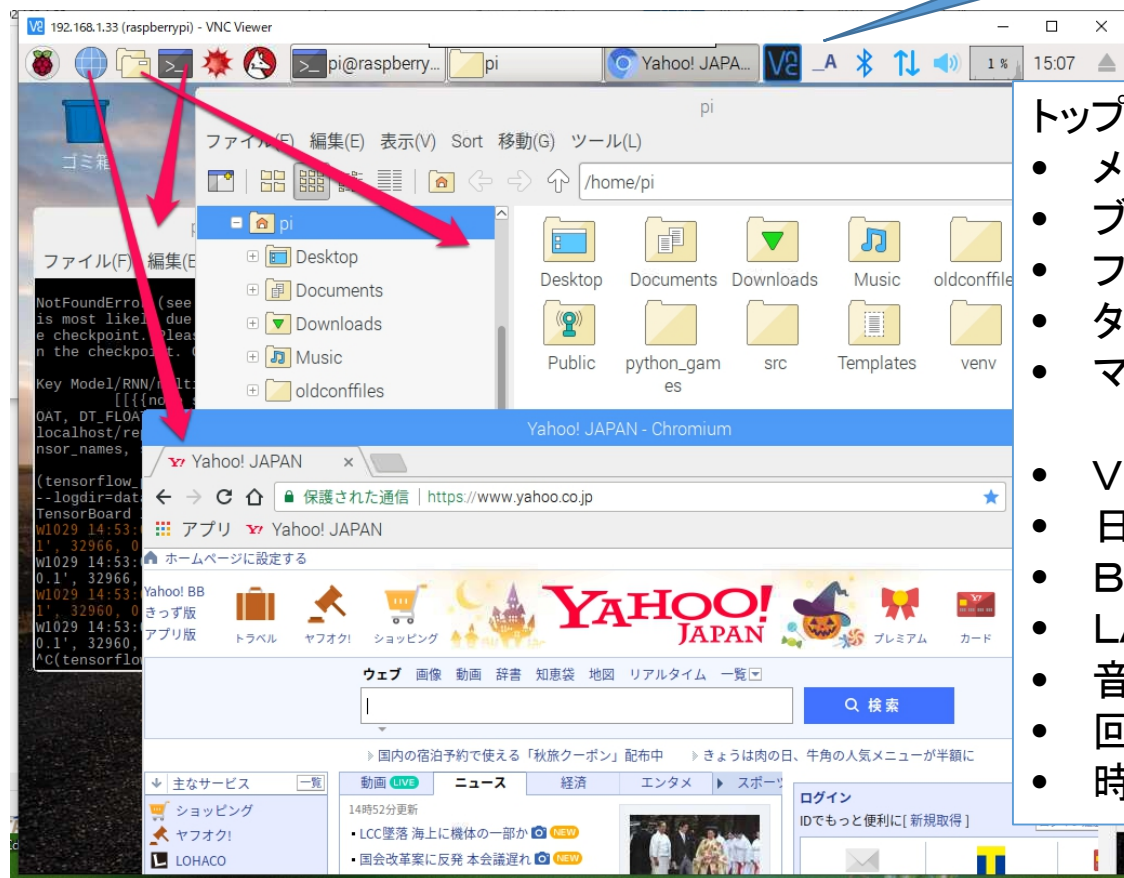
Raspberry Pi

Pi運用マニュアル

3. 基本操作

① 表示画面と内容

Anthyが出ない場合は、一度
Japaneseを選択後、再度
Anthyを選択してください。



トップ画面(上段のタスクバーで選択)

- メニュー
- ブラウザ
- ファイルマネージャ
- ターミナル
- マルチ画面選択

- VNC
- 日本語入力
- BLE
- LAN/WiFi
- 音量
- 回線効率
- 時刻

AWS IoT開発

②. AWS設定

• 手順1: アカウントの作成

- <https://aws.amazon.com/jp/register-flow/>
- 必要なメールアドレス、パスワードなどを入力します。

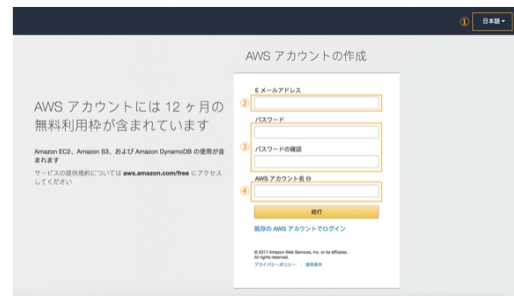


AWS アカウント作成の流れ

AWS アカウントを作成すると、1年間の無料利用枠はもちろん、AWS クラウドの世界中のリージョンで提供されるすべてのサービスを始めることができます。こちらでは日本のお客様に AWS アカウント作成におけるポイントをご紹介します。



ステップ 1: AWS アカウントの作成



※クリックすると大きな画像でご覧いただけます。

このページの上部タイトルおよび、末尾に設置されているオレンジ色のアカウント作成ボタンよりサインアップ画面へ移動します。

各ページ右上 ① の言語選択ボックスより、「日本語」でない場合「日本語」を選択後、こちらのサインアップ画面へお進みください。

最初に AWS アカウントとなる情報を設定します。

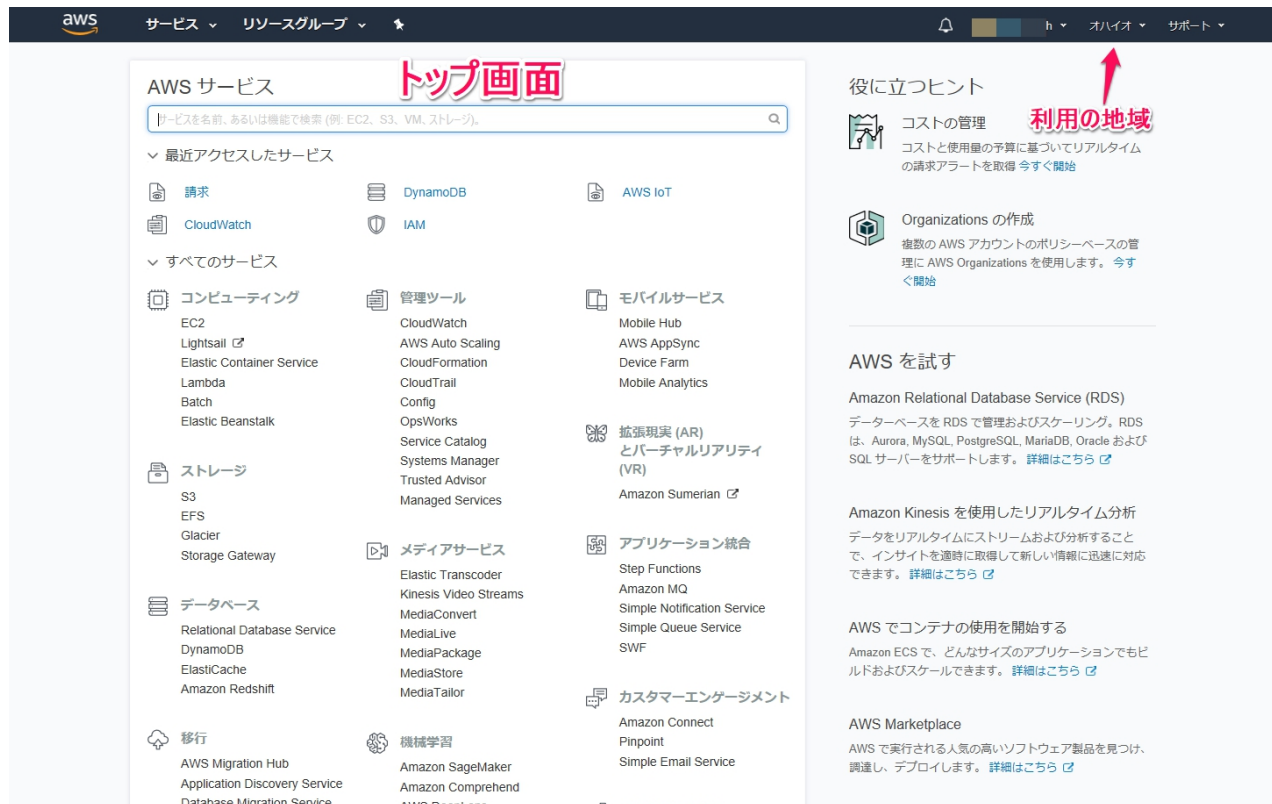
- ② の「E メールアドレス」には、AWS へのログインに利用したいメールアドレスを設定します。（※）
- ③ の「パスワード」および「パスワードの確認」で AWS へのログイン時に使用するパスワードを設定し、さらに確認用にもう一度同じパスワードを入力します。
- ④ の「AWS アカウント名」テキストボックスに、お客様のお名前を半角アルファベットで入力します。
- 入力後、「続行」ボタンをクリックします。

※ご登録いただくメールアドレスは、AWS からの通知等にも利用されます。複数の方へ

AWS IoT開発

②. AWS設定

- AWSトップ画面
 - 利用する場合に、地域を意識して設定してください。地域毎に料金が変わったり、利用できるサービスが限定されている場合があります。



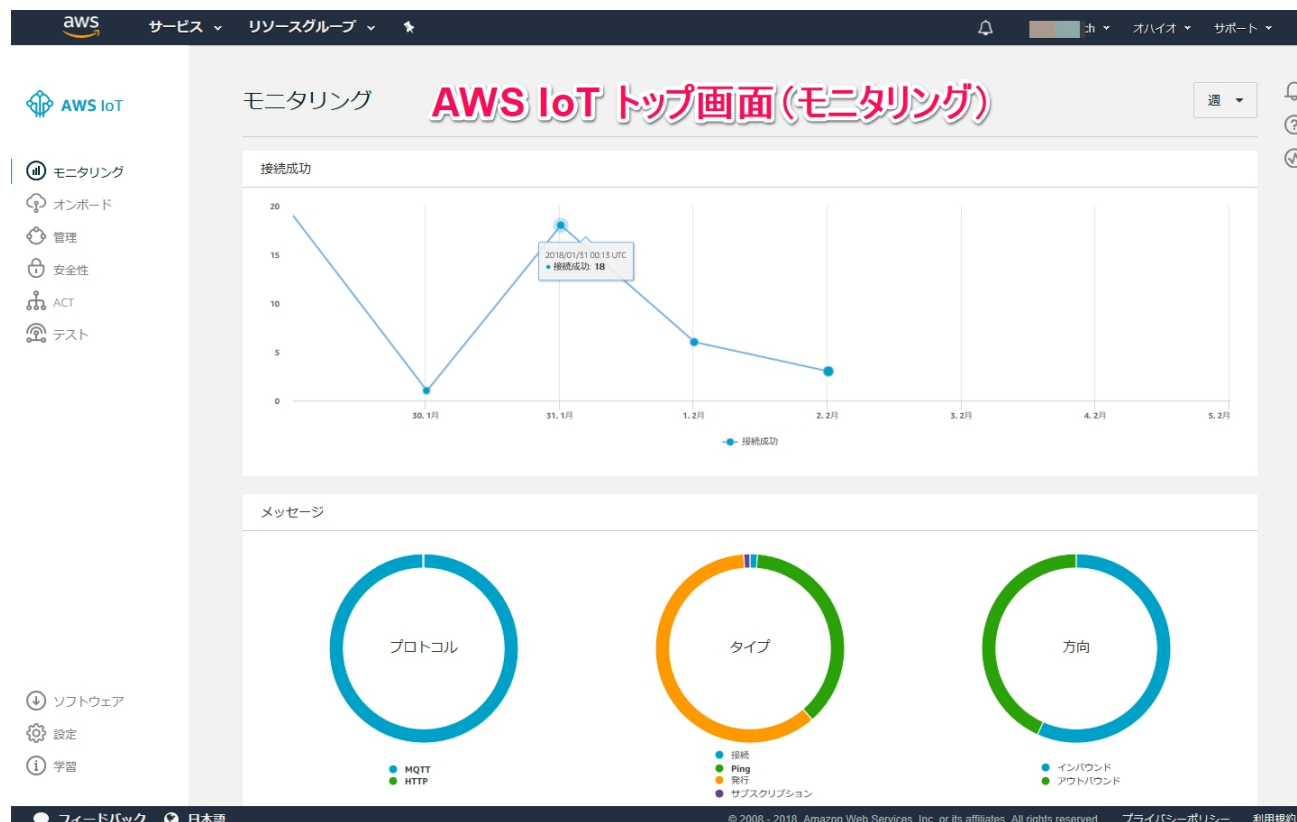
③. AWS IoT設定



AWS IoT Core

• 画面説明1

- AWSTップ画面からAWS IoTのサービスを選択します。
- モニタリング: 接続状況がひとめで分かります

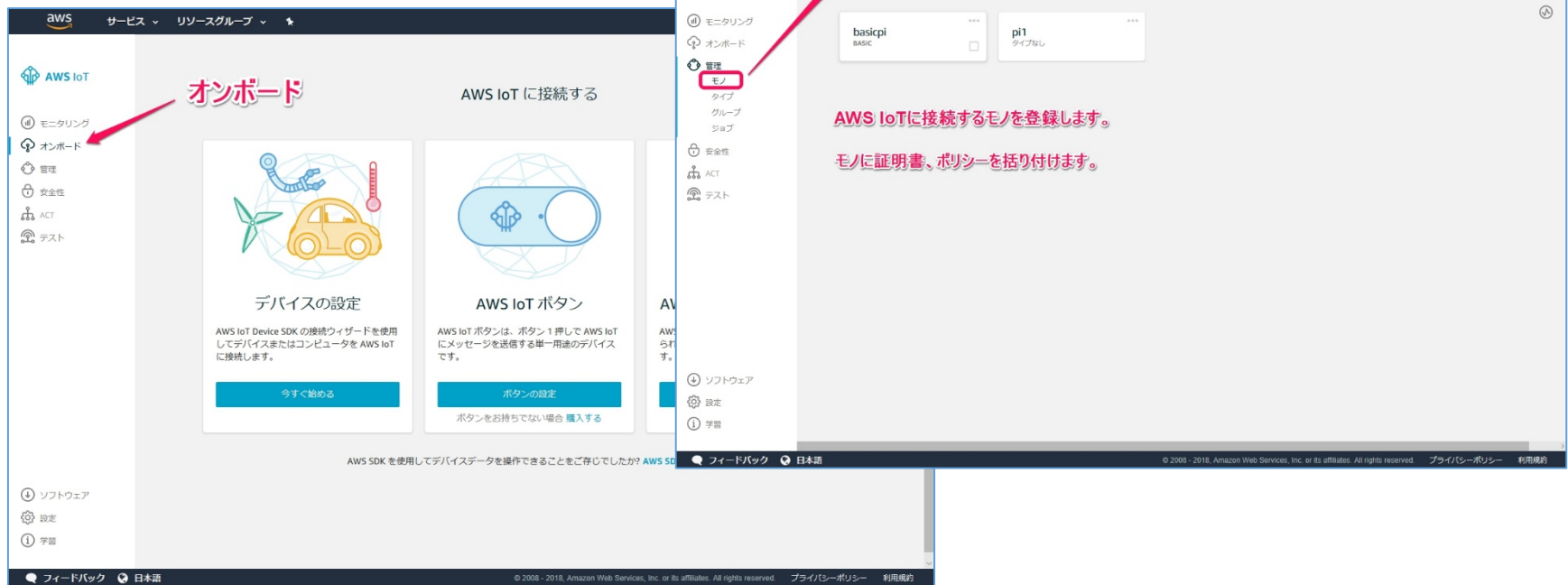


③. AWS IoT設定



AWS IoT Core

- 画面説明2
 - オンボード: 特に使用しません
 - 管理: モノ



オンボード

モノ

AWS IoTに接続する

デバイスの設定

AWS IoT ボタン

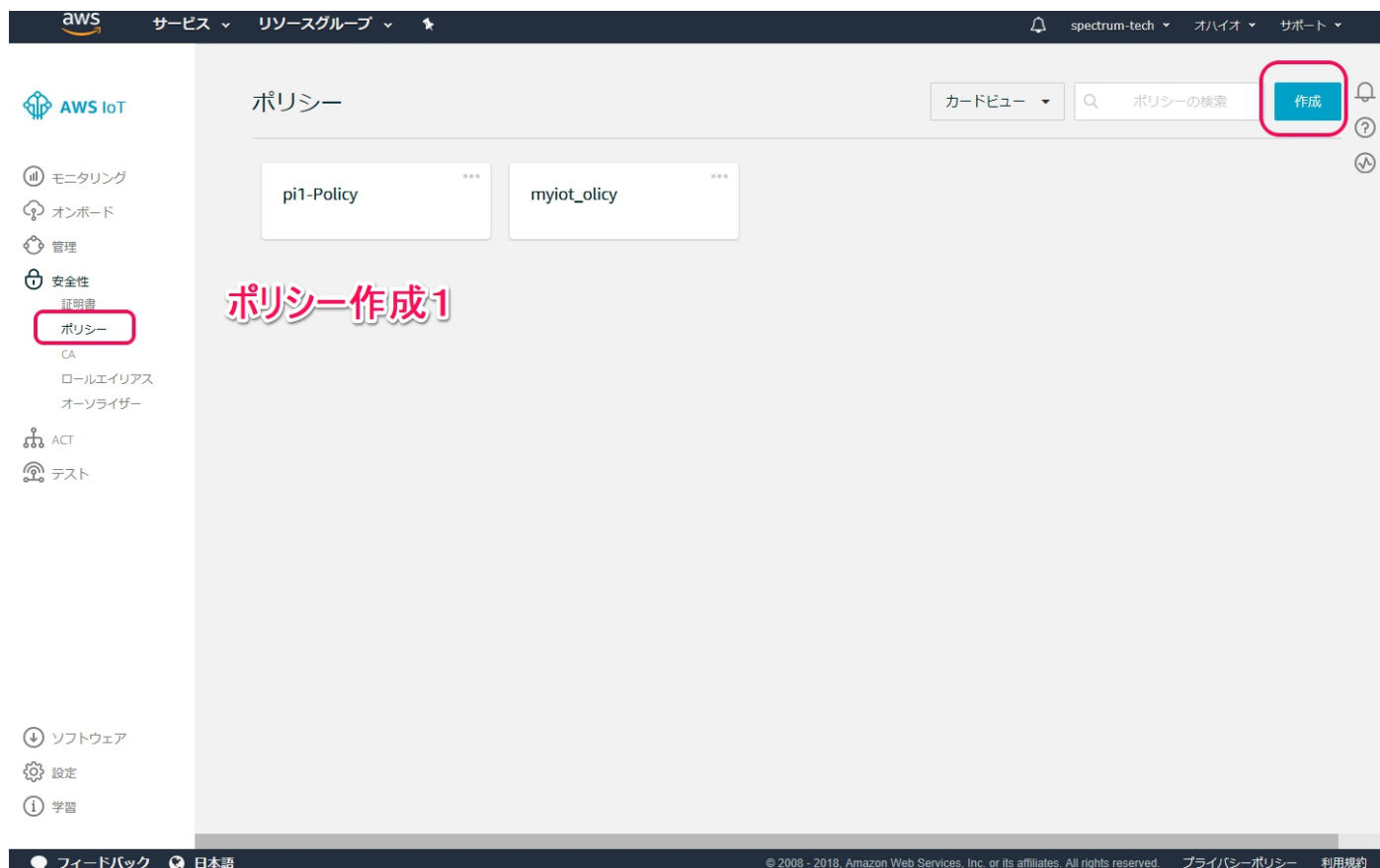
モノを登録します。

モノに証明書、ポリシーを割り付けます。

AWS IoT開発

③. AWS IoT設定

- ポリシー作成1
 - 安全性>ポリシーの作成を押します。

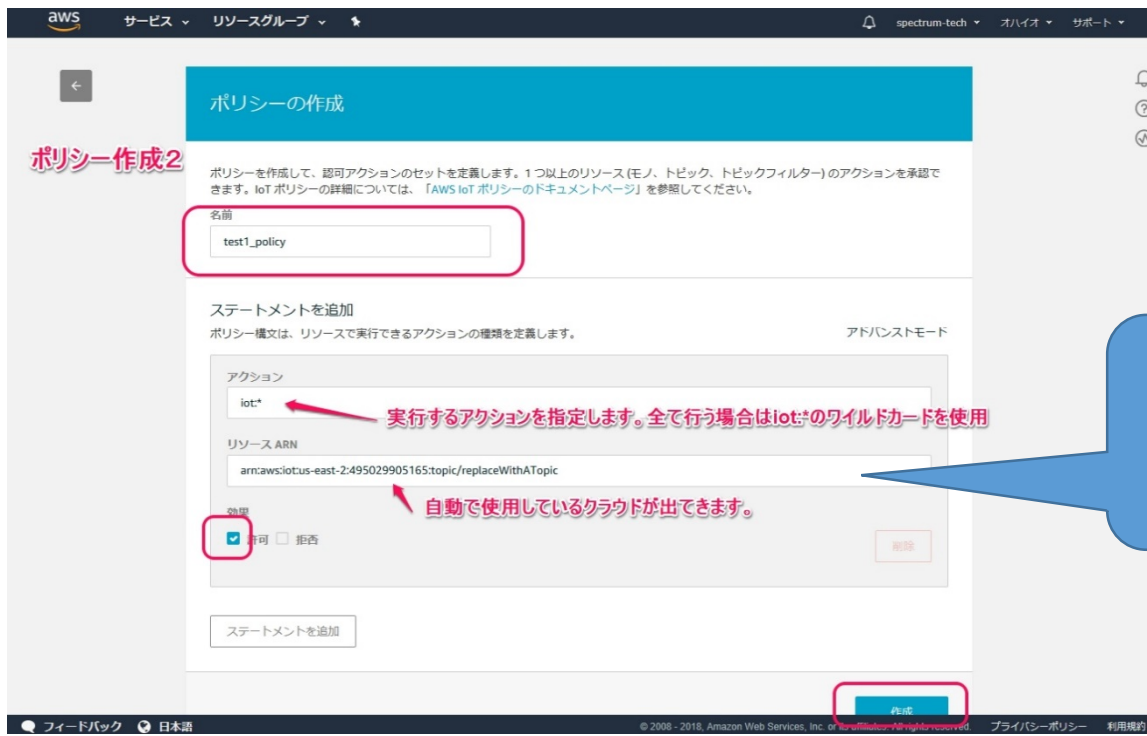


AWS IoT開発

③. AWS IoT設定

• ポリシー作成2

- 名前を入力します。
- アクション: `iot:connection`などの実行するアクションを入れます。ワイルドカードも使用できます。慣れるまで、`iot:*`と入力します。
- リソースARN: 現在利用しているクラウドのリソースが表示されます。そこからの接続のみに限定されます。又は*の無制限でも可能です。



ポリシーの作成

ポリシー作成2

ポリシーを作成して、認可アクションのセットを定義します。1つ以上のリソース (モノ、トピック、トピックフィルター) のアクションを承認できます。IoT ポリシーの詳細については、『AWS IoT ポリシーのドキュメントページ』を参照してください。

名前

test1_policy

ステートメントを追加

ポリシー構文は、リソースで実行できるアクションの種類を定義します。 アドバンスドモード

アクション

iot*

リソース ARN

arn:aws:iot:us-east-2:495029905165:topic/replaceWithATopic

効果

☒ 許可 ☐ 拒否

ステートメントを追加

作成

リソースARNを*
にしてください。

③. AWS IoT設定



AWS IoT Core

• ポリシー作成3

- 安全性>証明書>モノの登録で作成した証明書にポリシーをアタッチします。
- 右クリックして>ポリシーのアタッチを押します

証明書 **ポリシー作成3**

カードビュー

証明書の検索

作成

証明書ID	ステータス
259e1c5de191ddb98...	アクティブ
a4b75954229e44f37...	アクティブ
ef0b8f9919c64310f4...	アクティブ
235c44e7435ab45d8...	アクティブ
c1ab54dd2198105e1...	アクティブ
587e63...	アクティブ

有効化
無効化
取り消し
転送を許可する
転送を拒否する
転送を取り消す
ポリシーのアタッチ
モノをアタッチする
ダウンロード
削除

モノの登録で作成した証明書にポリシーをアタッチします。
右クリック>ポリシーのアタッチ

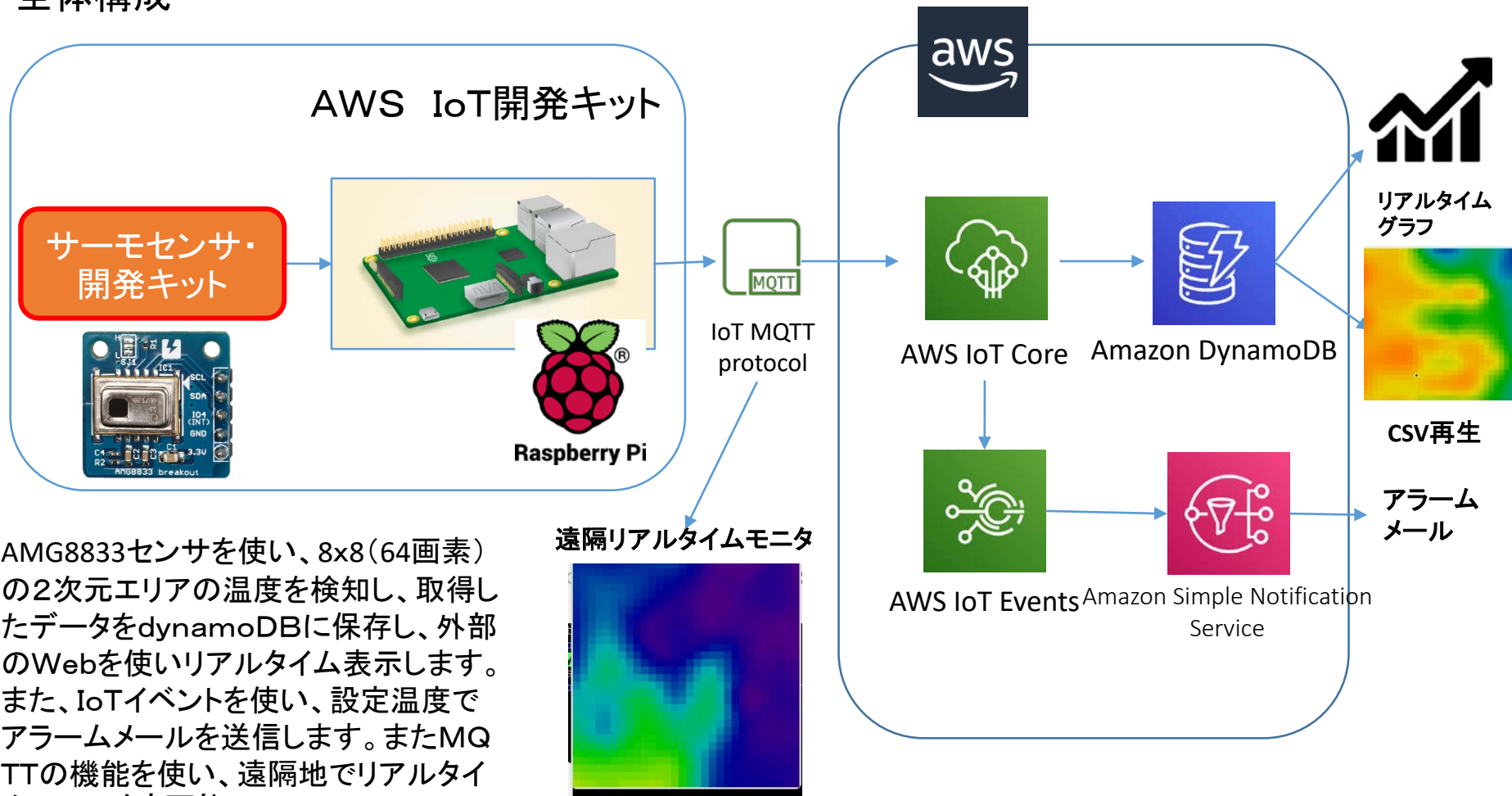
AWS IoT開発

④サーモセンサ情報収集、アラーム送付



AWS IoT Core

全体構成



AWS IoT開発

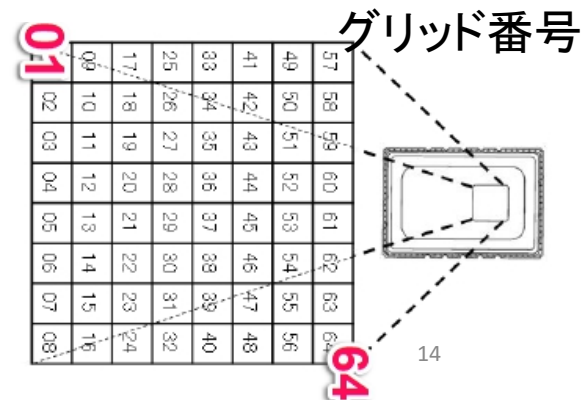
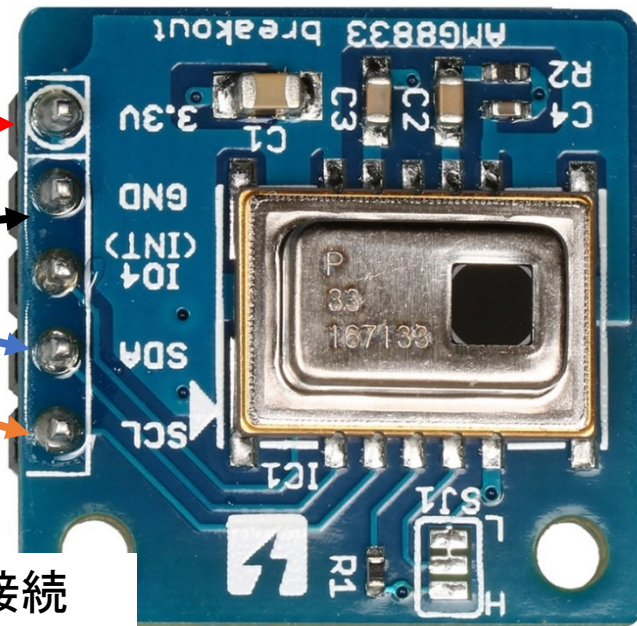
④サーモセンサ情報収集、アラーム送付

Raspberry PiとAMG8833の接続

J8			
3.3V	1	2	3
BCM2	3	4	5
BCM3	5	6	7
BCM4	7	8	9
Ground	9	10	11
BCM17	11	12	13
BCM27	13	14	15
BCM22	15	16	17
3.3V	17	18	19
BCM10	19	20	21
BCM9	21	22	23
BCM11	23	24	25
Ground	25	26	27
	27	28	29
BCM5	29	30	31
BCM6	31	32	33
BCM13	33	34	35
BCM19	35	36	37
BCM26	37	38	39
Ground	39	40	

メスオスのジャンパで接続
Pi ブレッドボード AMG8833

1-----+-----VDD
3-----SDA
5-----SCL
9-----GND





AWS IoT開発

④サーモセンサ情報収集、アラーム

AMG8833単体試験

- Piコンソールでpython pixels_test.pyを動作させます。
- 数字の羅列が出力されます。
- 出力されない場合は、I2Cの設定を確認。0x68に値が表示されれば正常。

\$ i2cdetect -y 1

```
# cd  
/home/pi/Documents/Adafruit_AMG88xx_p  
ython/examples  
# python python pixels_test.py
```

```
pi@raspberrypi: ~/Documents/A...ruit_AMG88xx_python/examples _ □ X  
ファイル(F) 編集(E) タブ(T) ヘルプ(H)  
^CTraceback (most recent call last):  
  File "thermal_cam.py", line 76, in <module>  
    pygame.draw.rect(lcd, colors[constrain(int(pixel), 0, COLORDEPTH- 1)], (disp  
layPixelHeight * ix, displayPixelWidth * jx, displayPixelHeight, displayPixelWid  
th))  
KeyboardInterrupt  
pi@raspberrypi:~/Documents/Adafruit_AMG88xx_python/examples $ python thermistor_  
test.py  
Thermistor Temp = 21.94 *C  
pi@raspberrypi:~/Documents/Adafruit_AMG88xx_python/examples $ python pixels_test  
.py  
[17.75, 18.0, 17.75, 18.0, 17.75, 17.0, 18.0, 18.75, 17.5, 17.75, 17.5, 17.75, 1  
7.75, 17.25, 17.75, 18.0, 18.25, 18.0, 17.5, 17.75, 18.0, 17.25, 17.5, 18.5, 18.  
25, 17.75, 17.75, 17.5, 17.75, 17.25, 18.0, 18.0, 17.5, 18.0, 17.25, 17.75, 17.5  
, 17.75, 17.75, 18.25, 17.5, 17.75, 17.75, 17.75, 16.75, 17.5, 17.25, 17.75, 18.  
25, 18.0, 18.25, 17.5, 17.5, 17.0, 17.5, 17.0, 16.75, 18.25, 17.25, 16.75, 16.5,  
17.0, 18.0, 17.0]  
[17.5, 17.75, 18.0, 18.0, 18.5, 18.0, 18.0, 17.75, 17.5, 18.0, 17.75, 17.25, 17.  
75, 17.75, 18.0, 17.75, 18.0, 18.0, 17.75, 18.0, 17.5, 17.75, 17.75, 17.75, 17.7  
5, 17.25, 18.0, 18.0, 17.25, 17.0, 18.0, 17.5, 18.25, 17.75, 17.5, 18.0, 17.5, 1  
7.75, 17.0, 18.25, 18.0, 17.75, 18.0, 17.75, 17.25, 17.75, 17.75, 17.75, 17.75,  
17.75, 18.5, 17.5, 17.25, 16.75, 18.0, 18.0, 16.75, 18.5, 17.5, 17.0, 16.5, 17.0  
, 17.0, 17.25]  
[18.0, 17.75, 17.75, 18.25, 18.0, 17.5, 18.25, 18.25, 17.75, 18.25, 17.75, 17.5,
```



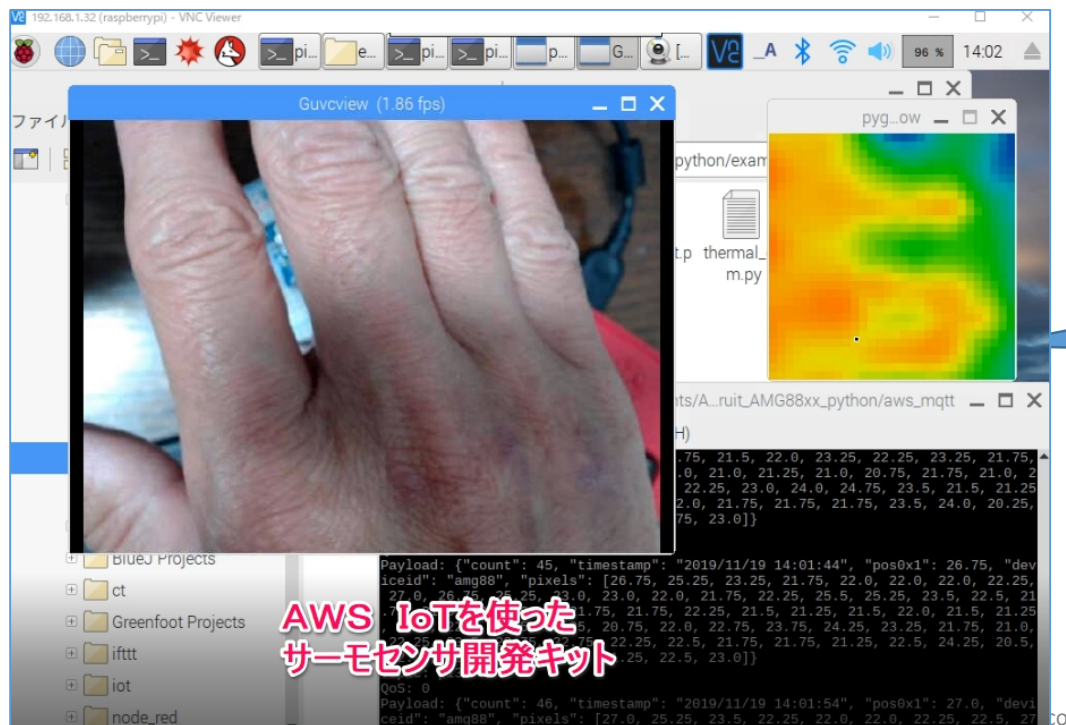

AWS IoT開発

④サーモセンサ情報収集、アラーム

AMG8833 画像試験

- Piコンソールでpython thermal_cam.pyを動作させます。
- 240x240ピクセルの画面で表示されます。
- プログラムで、最低:20度、最高:32度に設定しています。適宜数字を変えてください
- 現在の温度は、python thermistor_test.pyでテストしてください。

```
# cd  
/home/pi/Documents/Adafruit_AMG88xx_p  
ython/examples  
# python thermal_cam.py
```

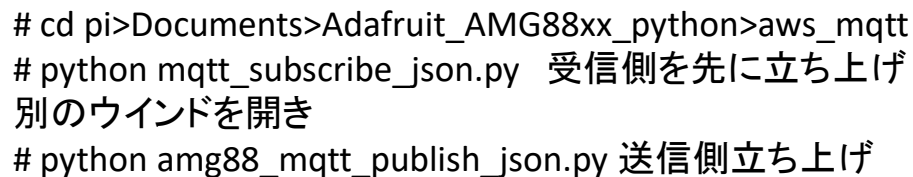


手をかざした
状態

**AWS IoTを使った
サーモセンサ開発キット**

AWS IoT折り返し試験

- Piのコマンド画面からsubscriber, publisherのやりとりを実施
- subscribe側に温度のデータ(リスト形式; 64個)が受信できていればOk, なおテスト用でpos0x1の最初のデータをグラフ表示のため取り出し



AWS IoT開発

④サーモセンサ情報収集、アラーム送付

リアルタイム表示

- IoT. Kyoto様の利用>グラフ
- Deviceid:amg88(pythonで書き込み済)
- 表示対象をチェックすると自動で表示されます。



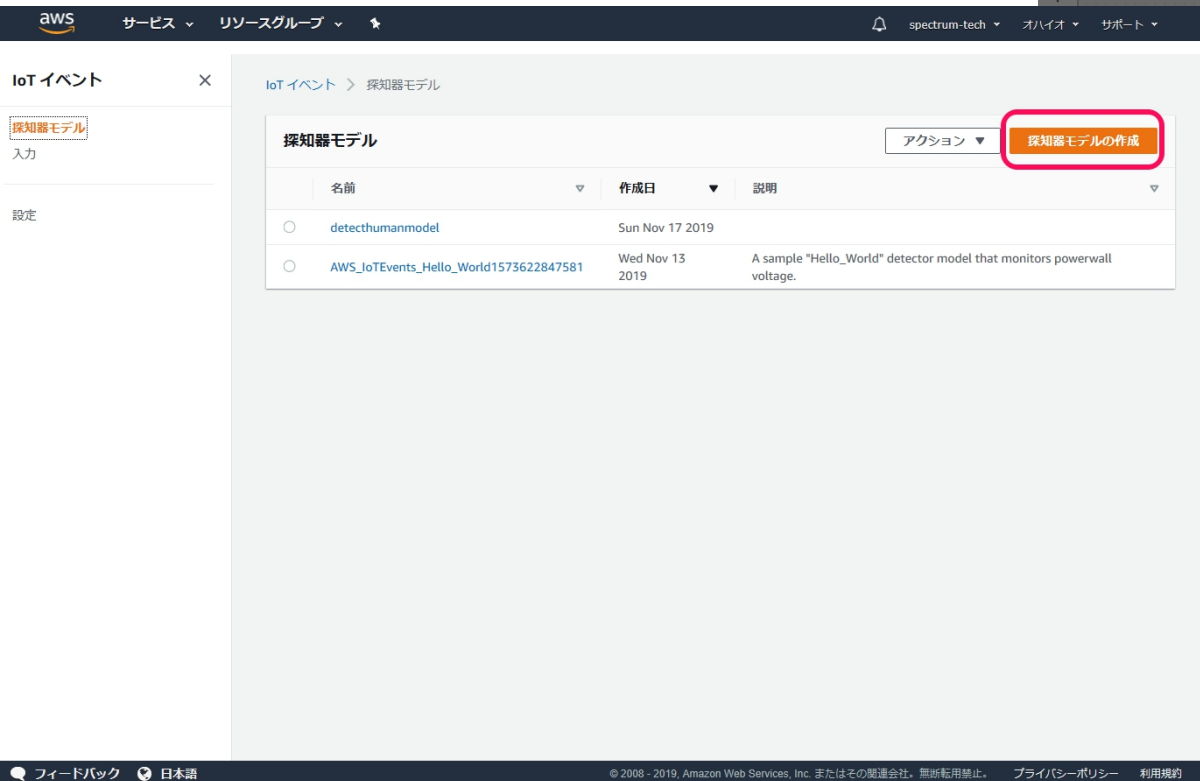
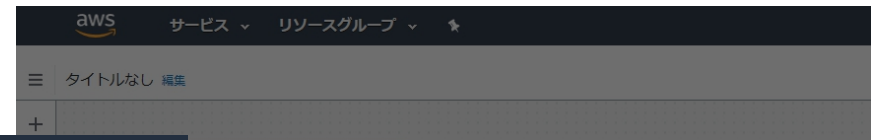
8x8=64個のデータの
うち、0x0のデータの
み表示。他はリスト
形式でDBに確保

AWS IoT開発

④サーモセンサ情報収集、アラーム送付

IoTイベント設定

- 指定設定温度で、メール送信するため、IoTイベントを設定します。
- 探知器モデル>探知器モデルの作成
- 新しく作成する



IoT イベント

探知器モデル

探知器モデルの作成

名前	作成日	説明
detecthumanmodel	Sun Nov 17 2019	
AWS_IoTEvents_Hello_World1573622847581	Wed Nov 13 2019	A sample "Hello_World" detector model that monitors powerwall voltage.



AWS IoT イベント

探知器モデル

作成方法

新しいディテクターモデルを作成

新しく作成する

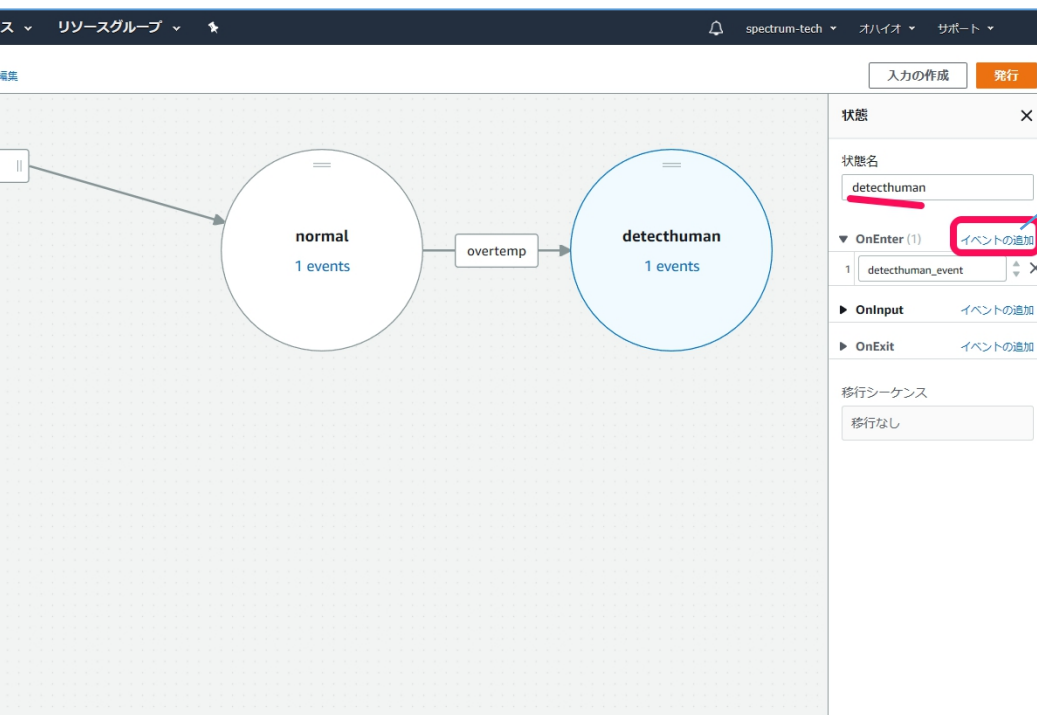
探知器モデルでの入力の使用

入力値は、アクシオンを起動したり状態を変更したりする入力値は、左のナビゲーションから利用できる【入力】ページで

④サーモセンサ情報収集、アラーム送付

IoTイベント設定

- 状態名: detecthuman
- イベントの追加
- イベント名: detecthuman_event
- イベント条件: `$variable.tempalarm==1`
- イベントアクション: SNSメッセージの送信



OnEnter イベントの追加

イベント名
detecthuman_event

イベントの条件 - オプション
探知器モデルがこのイベントをトリガーすることを知らせる入力およびその属性値を入力します。
`$variable.tempalarm==1`

イベントアクション
イベントがトリガーされたときに実行するアクションを選択します。
▶ SNS メッセージの送信 Remove

Add action

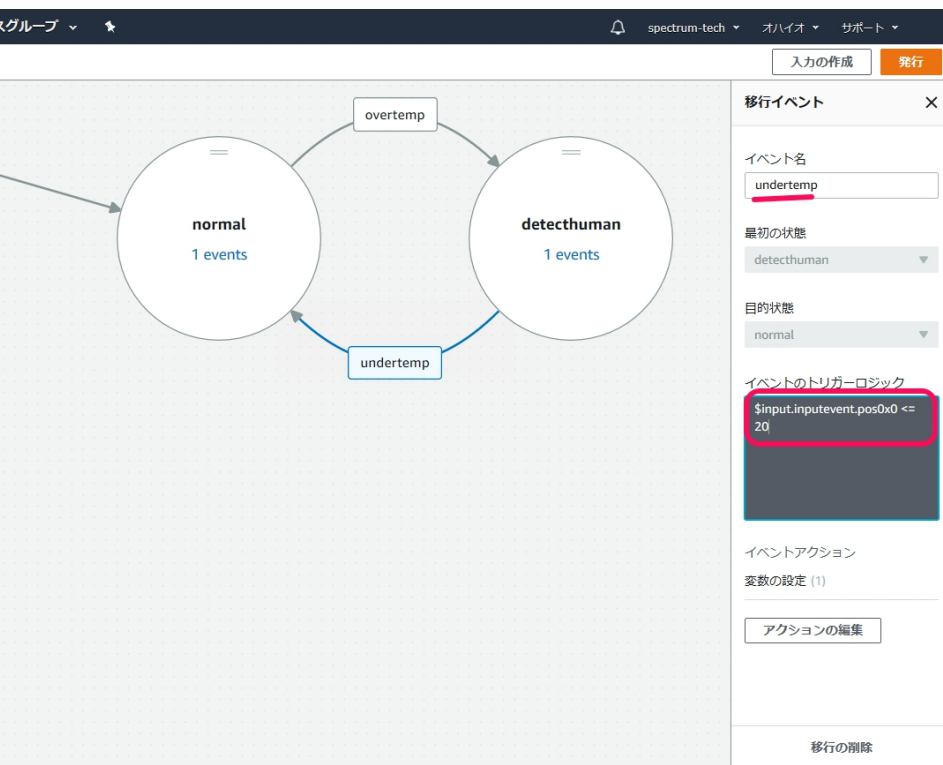
Cancel Save

AWS IoT開発

④サーモセンサ情報収集、アラーム送付

IoTイベント設定

- 移行イベントの作成
- イベント名: undertemp
- トリガーロジック: `$input.inputevent.pos0x0 <= 20`
- 変数の設定 > 値の割り当て
- 変数名: tempalarm 値: 0



移行イベントアクションの編集

イベントアクション

イベントがトリガーされたときに実行するアクションを選択します。

▼

変数の設定

Remove

▲▼

変数オペレーション

この変数の値が変更される方法を選択します。

値の割り当て

▼

変数名

既存の変数の名前を指定するか、新しい変数の名前を指定して作成します。

tempalarm

値の割り当て

変数に手動で値を割り当てます。

0

Add action

Cancel

Save

温度が20度を超えたら
tempalarmを1に

spectrum technology co.

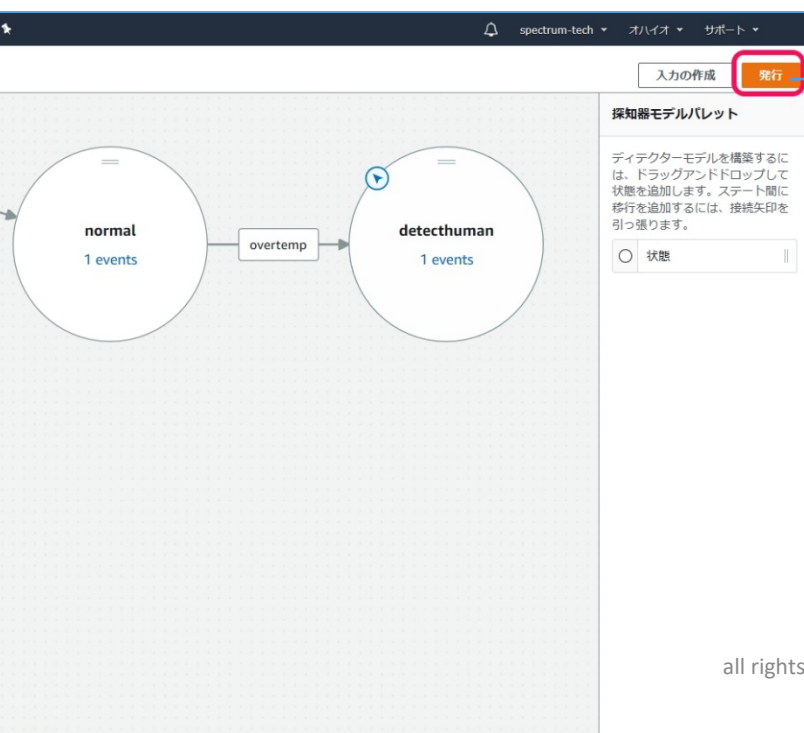
21

AWS IoT開発

④サーモセンサ情報収集、アラーム送付

IoTイベント設定

- 探知器モデルの発行
- モデル名: detecthumanmodel
- ロール: myevent_role 自動設定されます。
- 単一の探知器作成、シリアル評価
- 保存して発行。エラーが出なければ作成完了



探知器モデルの公開

探知器モデル名
ディテクターモデルに、後から識別しやすい名前をつけます。

detecthumanmodel
A-Za-z0-9_ から 1~128 文字を使用します。

説明 - オプション
説明は 128 文字より長くすることはできません。

ロール
探知器モデルにロールを追加します。AWS IAM から既存のロールを選択します。 [AWS IAM ロールの詳細](#)

myevent_role

探知器生成メソッド
探知器モデルのバージョンを発行すると、1 つ以上の探知器が自動的に作成されます。次のオプションでは、作成する探知器の数を決定できます。探知器が作成される方法を選択します。

☐ 一意のキー値ごとに探知器を作成する
この探知器モデルが多数のデバイスあるいはプロセスを追跡する場合には、この設定を使用します。

☒ 単一の探知器を作成する
この探知器モデルが単一のデバイスセットあるいはプロセスセットを追跡する場合には、この設定を使用します。

ディテクターの評価方法
評価方法とは、イベントが評価される順序とアクションの実行方法を指します。

☐ バッチ評価
変数が更新され、すべてのイベント条件が評価された後にのみイベントが実行されます。

☒ シリアル評価
変数が更新され、イベント条件はイベントが定義されている順序で評価されます。

キャンセル 保存して発行する

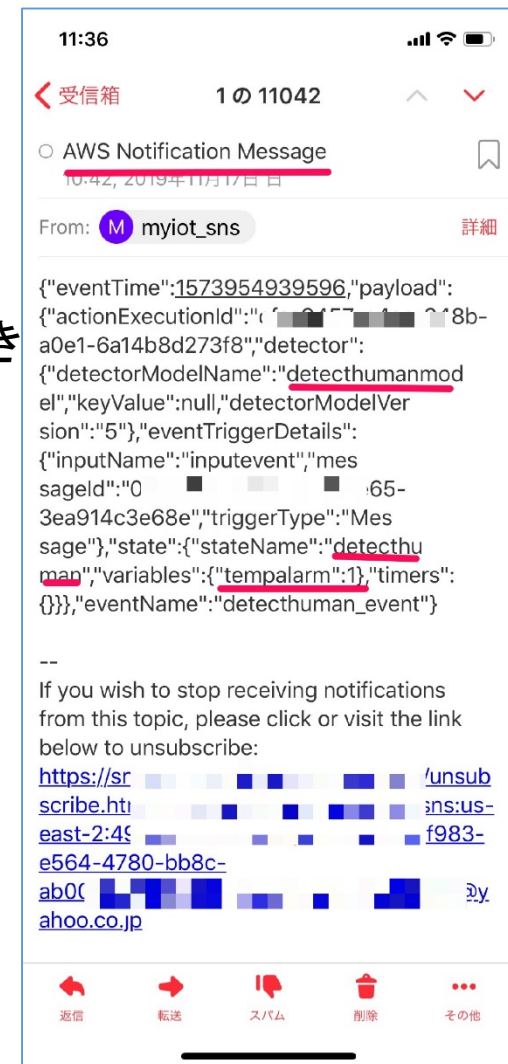
AWS IoT開発

④サーモセンサ情報収集、アラーム送付

AWS IoT Core

IoTイベント試験

- 探知器モデル>サンプルデータの送信
- 入力名:inputevent
- Pos0x0:25,番号と入力し、プレビュー
- データの送信
- メールの確認
 - AWS Notification Messageが受信できます



AWS IoT開発

④サーモセンサ情報収集、アラーム送付

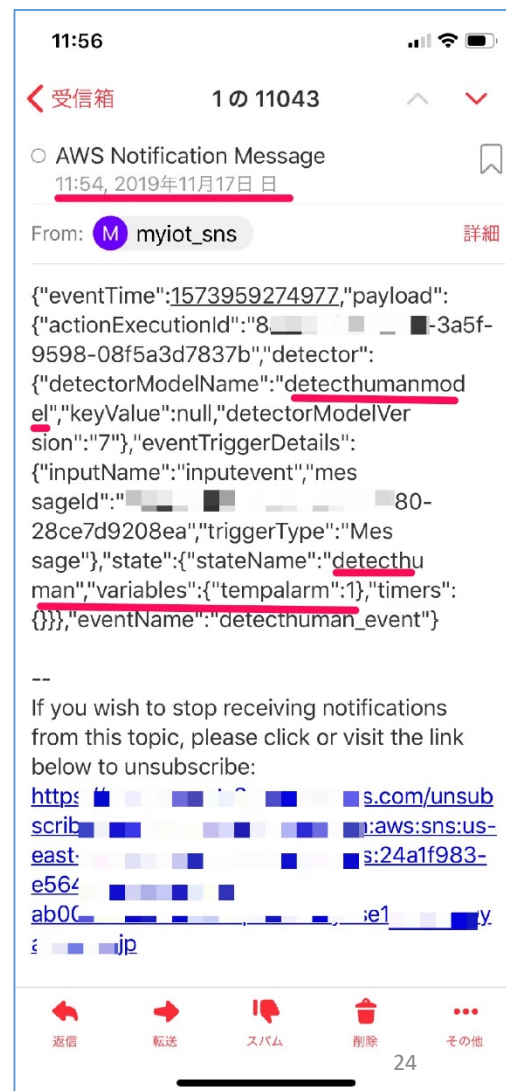
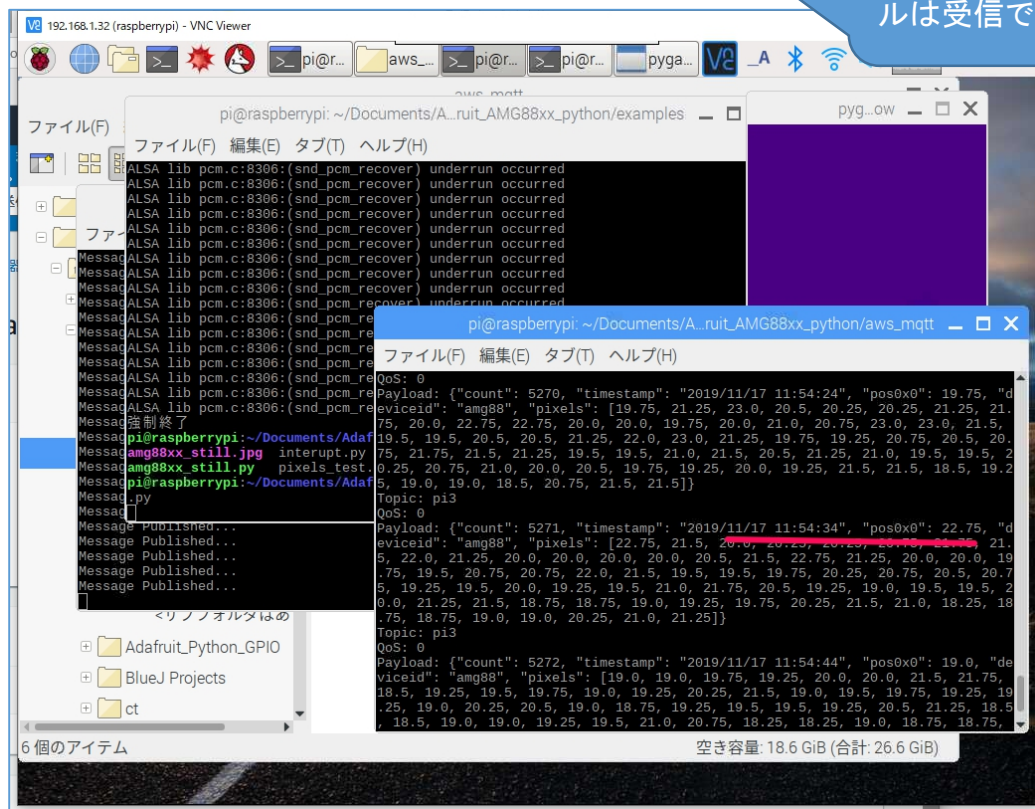


AWS IoT Core

IoTイベント試験

- Amg8833で手をかざして温度を20度以上にします。
- メールの受信を確認

イベントがNormalにもどっていないと、メールは受信できません。



AWS IoT開発

AWS IoT Core

④サーモセンサ情報収集、アラーム送付

```
$ cd  
/home/pi/Documents/Adafruit_AMG88xx_python/aws_mqtt  
$ python mqtt_subscribe_mon.py
```

遠隔リアルタイム・モニタ

- Mqttのsubscribe機能を使って、遠隔地でリアルタイムにサーモセンサの画像のモニタを行います。Mqttの送信タイミングの10秒毎受信
- `python mqtt_subscribe_mon.py`



AWS IoT開発

④サーモセンサ情報収集、アラーム送付

CSVの画像再生

- DynamoDBのCSVエクスポートしたoutput.csvを画像再生します
- `python thermal_csv.py`

```
$ cd  
/home/pi/Documents/Adafruit_AMG88xx_python/DynamoDB  
toCSV  
$ python thermal_csv.py
```

