ユーザーガイド

[MQTT ゲートウェイ編]

※MQTT ゲートウェイは試験段階の機能です

Ver. 1.18.0

【改訂履歴】

版	改訂内容	改訂日
1.18.0	・初版作成	2026/03/16

目次

1. はじめに

本書では、MQTT（IoT 向けのメッセージングプロトコル）を用いた UECS ネットワーク同士の連携手段として、Arsprout Pi にて MQTT ゲートウェイを使用する方法を説明します。使用の前提条件として、MQTT ゲートウェイのライセンスコードが必要となります。現在は試験版のため期間限定の無償ライセンスを発行します。将来的には有償ライセンスとなります。事前に Rasppberry Pi 基板の MAC アドレス情報をメールに記載して、当社サポート窓口(support@arsprout.co.jp) へライセンス発行申請を行ってください。MAC アドレスは Arsprout Pi を Raspberry Pi にインストールした後に、ノード設定画面で確認できます。Arsprout Pi のインストール方法など、本書で説明していない内容については「Arsprout Pi ユーザーガイド[基本機能編]」、および「Arsprout Air セットアップ・設置マニュアル」を参照してください。

2. 機器構成

MQTT ゲートウェイは MQTT ブローカー（MQTT メッセージの仲介サーバー）を介して動作します。MQTT ブローカーは Arsprout Pi から接続可能であれば、プライベートなネットワーク環境内に自前で用意したもの、インターネットあるいはクラウドサービス上に存在するものを問わず、自由に指定できます。後者の MQTT ブローカーに接続するためには Arsprout クラウドへの接続と同様に、モバイル回線や Ethernet を経由する必要があります。

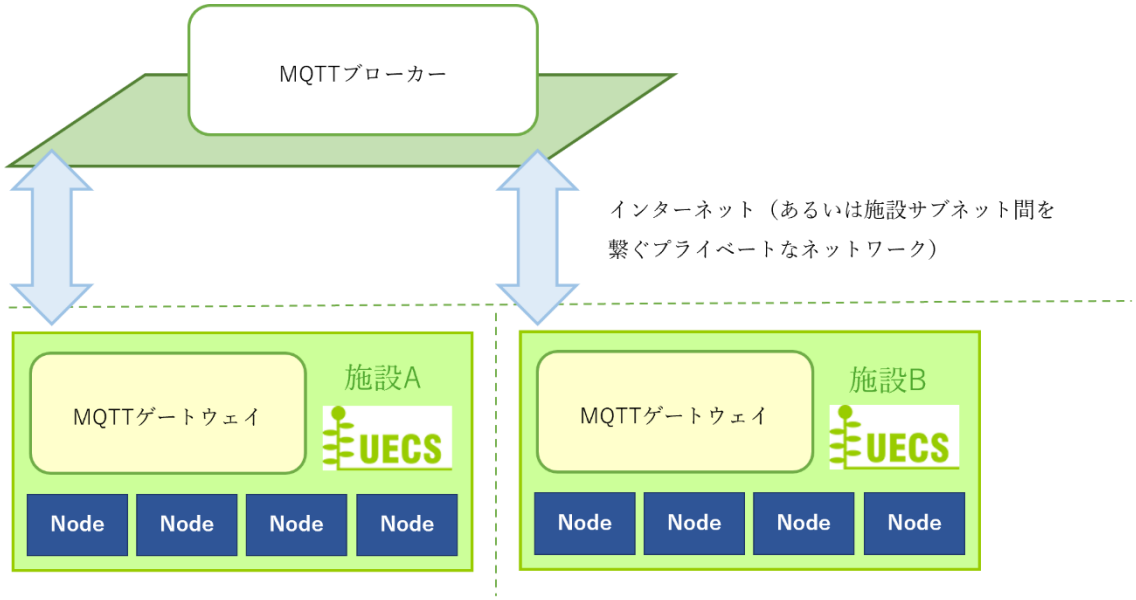


図 1：機器構成

3. ソフトウェア設定

Raspberry Pi とセットアップ用 PC を LAN ケーブルで接続し、Raspberry Pi の電源を ON にして 1 分待った後に、Arsprout Pi の管理画面（ http://<ノード IP アドレス> ）に WEB ブラウザでアクセスしてください。PC や Arsprout Pi の IP アドレス等の設定については、「Arsprout Pi ユーザーガイド[基本機能編]」を参照してください。

3.1. ライセンスコードの入力

① Arsprout Pi の「システムメニュー」->「ライセンスコード」を選択します。

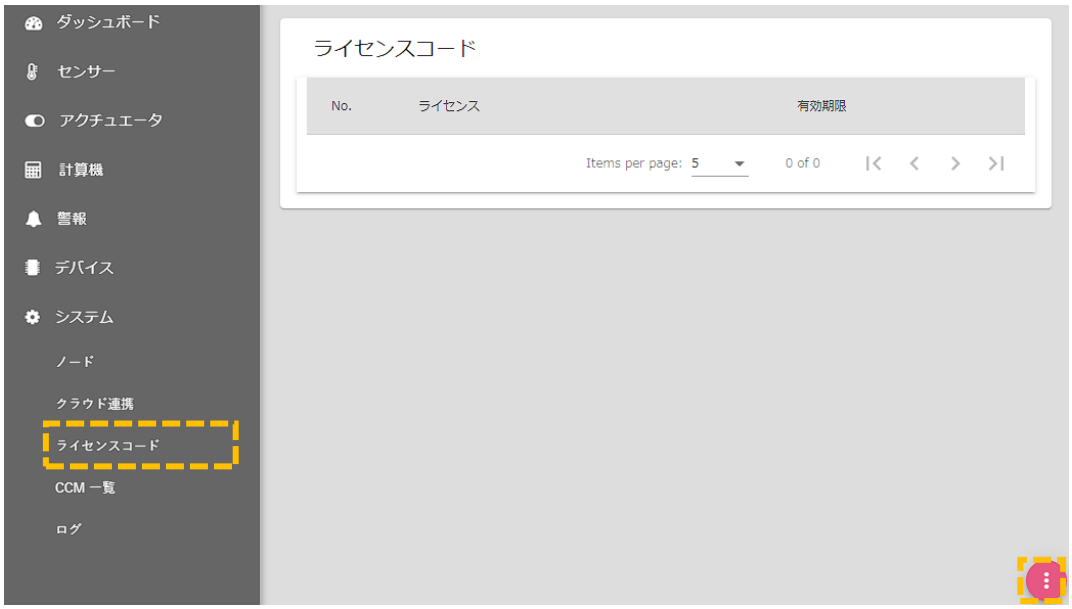


図 2：ライセンスコード一覧画面

② 「」ボタンをクリックすると、編集画面になります。

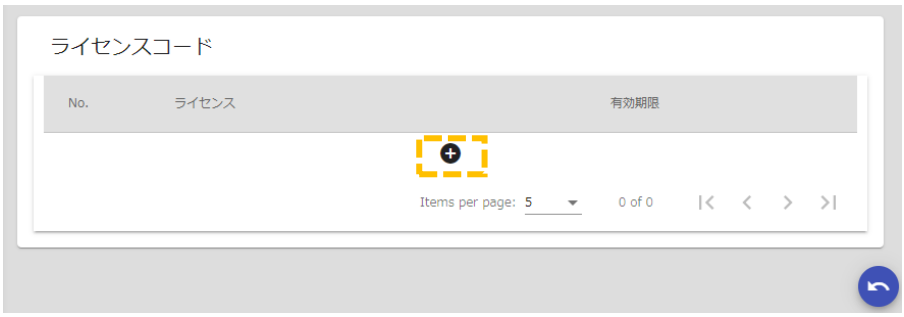


図 3：ライセンスコード編集画面

③　追加ボタン[⊕]をクリックし、ライセンス入力画面からライセンスコード入力を行い、送信ボタンで登録します。

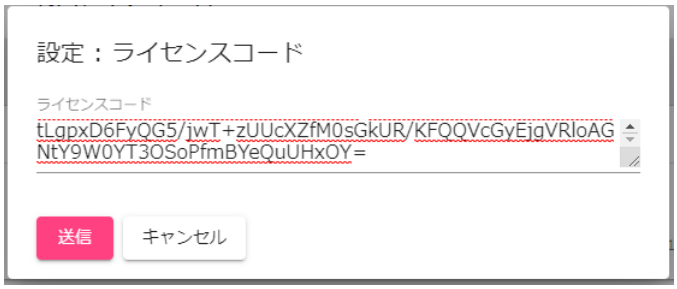


図 4：ライセンスコード入力ダイアログ画面

④　ライセンス一覧に以下のように表示されていれば完了です。

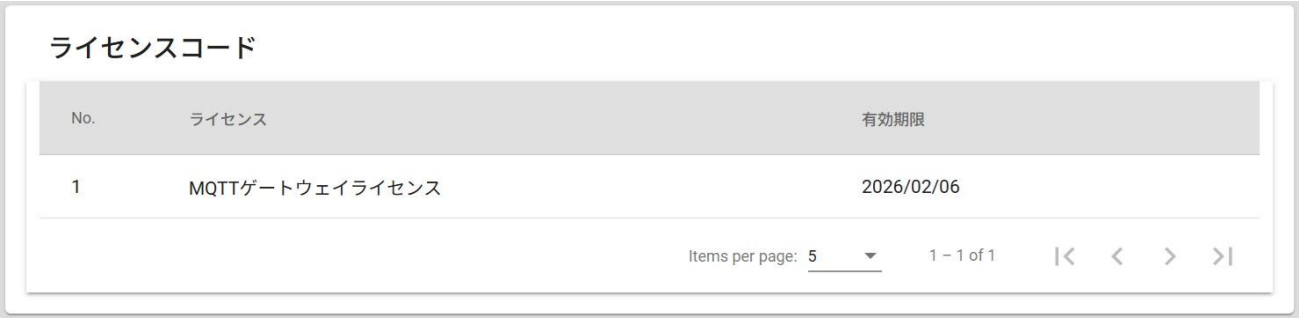


図 5：ライセンスコード確認画面

3.2. MQTT ゲートウェイの登録

① 「デバイス」メニューから「追加」を選択すると、新規デバイス作成ダイアログが表示されます。

図 6：デバイス新規追加画面

② 「通信インターフェース」カテゴリから「MQTT ゲートウェイ」を選択すると、メニューに新規デバイスが追加され、デバイス情報画面が表示されます。

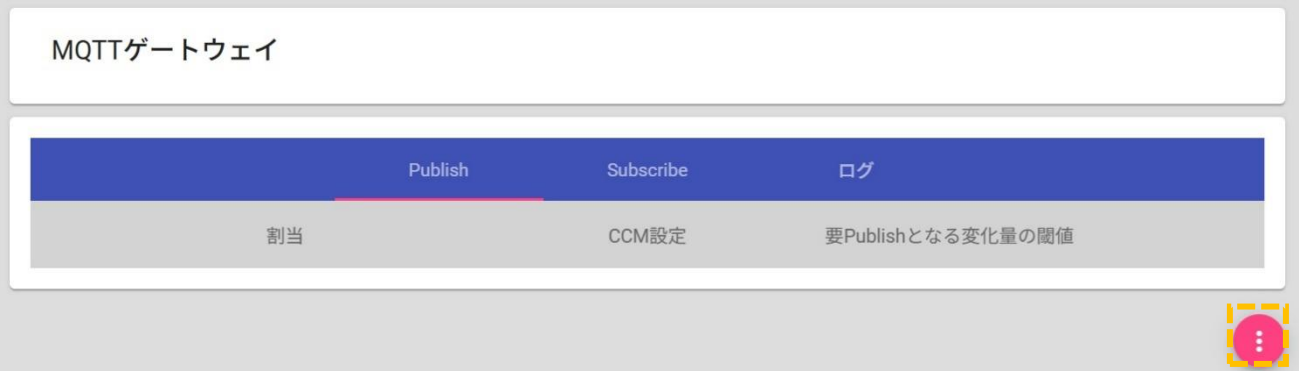


図 7：デバイス(Air ゲートウェイ)画面

③ 作成直後は MQTT ブローカーの設定等が入力されていないため、デバイスは稼働しません。右下の編集ボタン[⋮]をクリックして基本設定を行う必要があります。

図 8：デバイス設定(MQTT ゲートウェイ)画面

カテゴリ	項目	説明
基本設定	デバイス名	デバイスメニューに表示される名称です。識別しやすい名前に変更可能です。
	MQTT ブローカー URL	MQTT ブローカーの URL です。MQTT ブローカーを提供しているサービスとしては AWS IoT Core が有名です。手早く MQTT ブローカーを試したい場合は次章「MQTT ブローカーの準備」を参照してください。将来的には Arsprout クラウド公式の MQTT ブローカーの提供を予定しています。
	MQTT クライアント ID	MQTT ブローカーに接続するためのクライアント ID です。後述のユーザー名とは異なる概念であり、MQTT ブローカーによっては指定すべき内容が規定（例：AWS IoT Core ではモノの名前）されています。これに準拠しない場合、通信そのものに問題はなくとも、MQTT ブローカーの運用管理上の制約が生じる可能性があります。 適正ではない内容を指定したり、同じ内容の使いまわしを行うと、通信そのものの問題が生じることがあるため、指定すべき内容が不明な場合は空白のままとして下さい。空白とした場合は内部で極力安全なランダム文字列を生成して接続します。Arsprout クラウド公式の MQTT ブローカー（提供時期未定）を使用する場合、ここで指定する内容は当社サポートより通知されます。
	ユーザー名	MQTT ブローカーに接続するためにユーザー名です。
	パスワード	MQTT ブローカーに接続するためのパスワードです。
	データ記録環境名	MQTT ブローカーとデータをやりとりする際のトピック名（巻末付録にて補足）として使用されます。特別な意図がない限り、初期値の「prod」から変更する必要はありません。半角英数字とハイフン（-）とアンダースコア（_）のみ使用できます。

	データ所有者名	MQTT ブローカーとデータをやりとりする際のトピック名として使用されます。Arsprout クラウド公式の MQTT ブローカー（提供時期未定）を使用する場合、ここで指定する内容は当社サポートより通知されます。それ以外の場合、任意の名称です。半角英数字とハイフン（-）とアンダースコア（_）のみ使用できます。
	ルート CA 証明書	MQTT ブローカーと暗号化通信を行う際に使用します。「サーバー証明書」や「MQTT ブローカーの証明書」と表記されることもあります。この証明書を指定することにより、この MQTT ゲートウェイから見た、MQTT ブローカーとの通信の安全性を担保（接続先の誤りがない、暗号化手続きの不備がない等）できます。
	デバイス証明書	MQTT ブローカーと暗号化通信を行う際に使用します。「クライアント証明書」と表記されることもあります。この証明書を指定することにより、MQTT ブローカーから見た、この MQTT ゲートウェイとの通信の安全性を担保できます。
	秘密鍵	MQTT ブローカーと暗号化通信を行う際に使用します。デバイス証明書を指定する場合は、秘密鍵も指定する必要があります。

表 1：デバイス設定（MQTT ゲートウェイ）基本設定項目

④　右下の保存ボタン[ ]をクリックすると設定が保存されます。

3.3. Publish 設定の登録

① MQTT ゲートウェイ画面右下の編集ボタン[ ]をクリックして編集モードにします。

図 9：デバイス(MQTT ゲートウェイ)画面

② Publish タブにて中央下の「⊕」ボタンをクリックすると、Publish 設定が追加されます。Publish 設定により MQTT ブローカーにどのようなメッセージを Publish するのかについては巻末付録を参照してください。

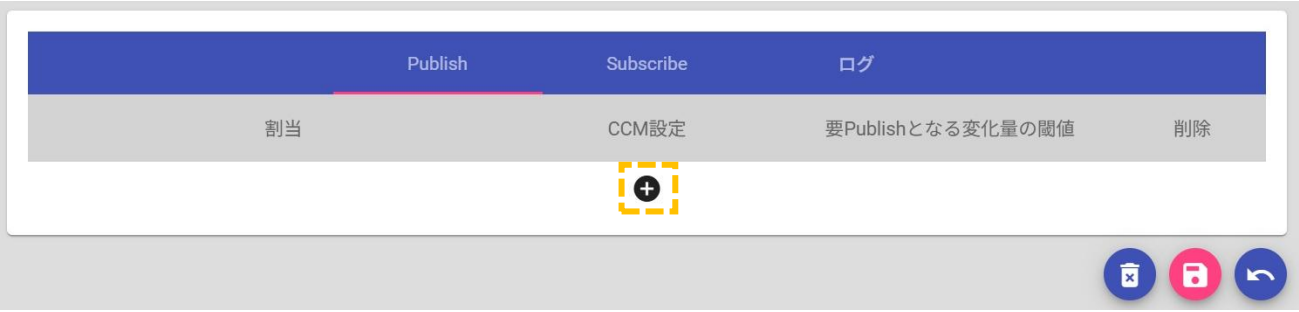


図 10：デバイス設定(MQTT ゲートウェイ)画面

図 11：Publish 設定

カテゴリ	項目	説明
Publish 設定	コンポーネント割当	コンポーネント（センサー/アクチュエータ等）を選択します。
	CCM 設定	コンポーネントがアクチュエータの場合、運転状態(opr)または遠隔制御(rcA)のいずれかを選択します。
	要 Publish となる変化量の閾値	コンポーネントの微細すぎる変化を毎回 Publish すると、通信量/ネットワーク負荷が増大します。これを防ぐために変化量の閾値を設定します。

表 2：Publish 設定項目

③ Publish 設定分「⊕」ボタンで順次追加していきます。

図 12：Publish 設定

④ 最後に、右下の保存ボタン[🔒]をクリックすると設定が保存されます。保存ボタンをクリックしないと設定が反映されませんので、忘れずにクリックしてください。

3.4. Subscribe 設定の登録

① MQTT ゲートウェイ画面右下の編集ボタン[⋮]をクリックして編集モードにします。

図 13：デバイス(MQTT ゲートウェイ)画面

② Subscribe タブにて中央下の「⊕」ボタンをクリックすると、Subscribe 設定が追加されます。Subscribe 設定により MQTT ブローカーからどのようなメッセージを Subscribe するのかについては巻末付録を参照してください。

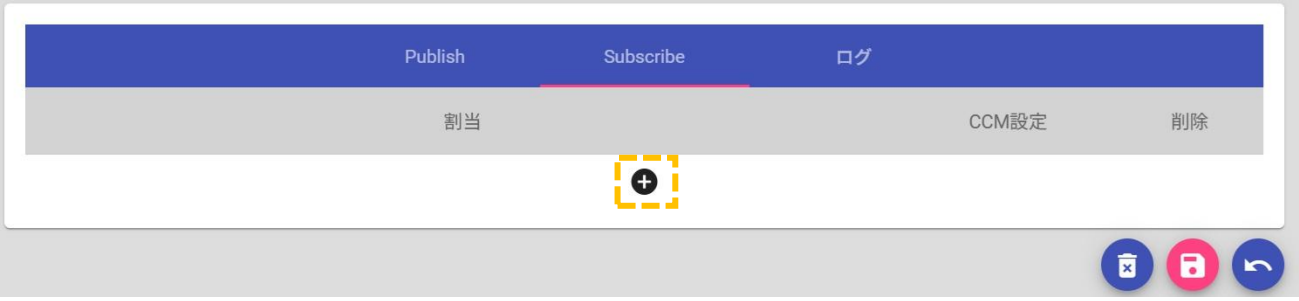


図 14：デバイス設定(MQTT ゲートウェイ)画面

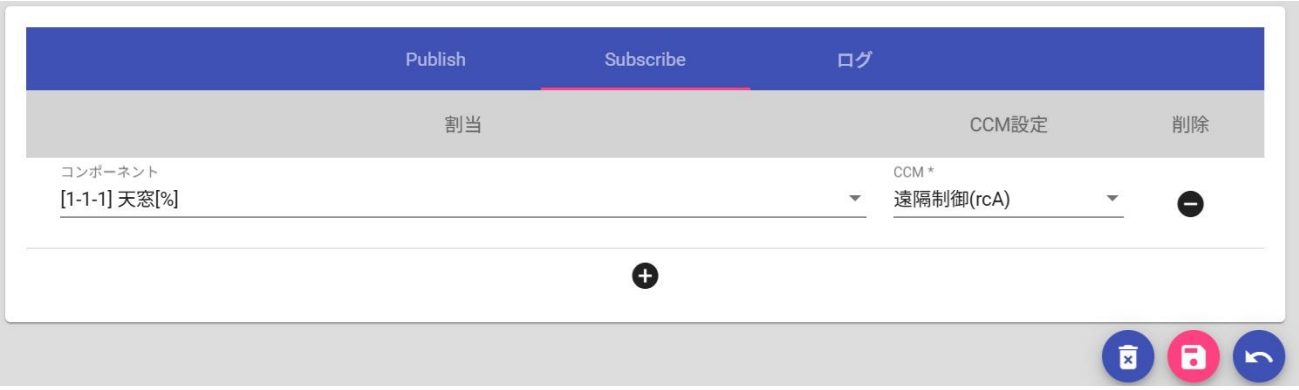


図 15：Subscribe 設定

カテゴリ	項目	説明
Subscribe 設定	コンポーネント割当	コンポーネント（センサー/アクチュエータ等）を選択します。
	CCM 設定	コンポーネントがアクチュエータの場合、運転状態(opr)または遠隔制御(rcA)のいずれかを選択します。

表 3：Subscribe 設定項目

③ Subscribe 設定分「⊕」ボタンで順次追加していきます。

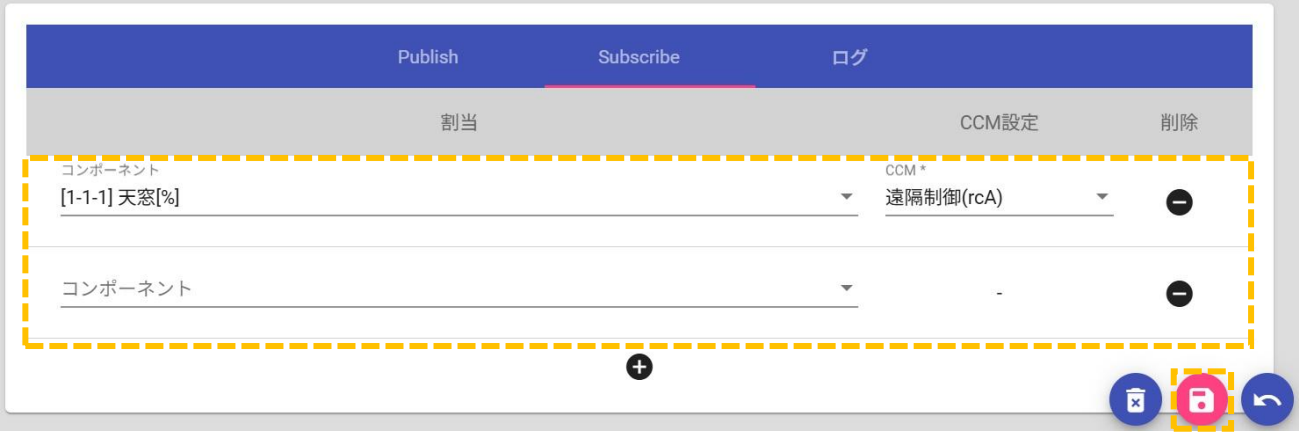


図 16：Subscribe 設定

④ 最後に、右下の保存ボタン[🔒]をクリックすると設定が保存されます。保存ボタンをクリックしないと設定が反映されませんので、忘れずにクリックしてください。

4. MQTT ブローカーの準備

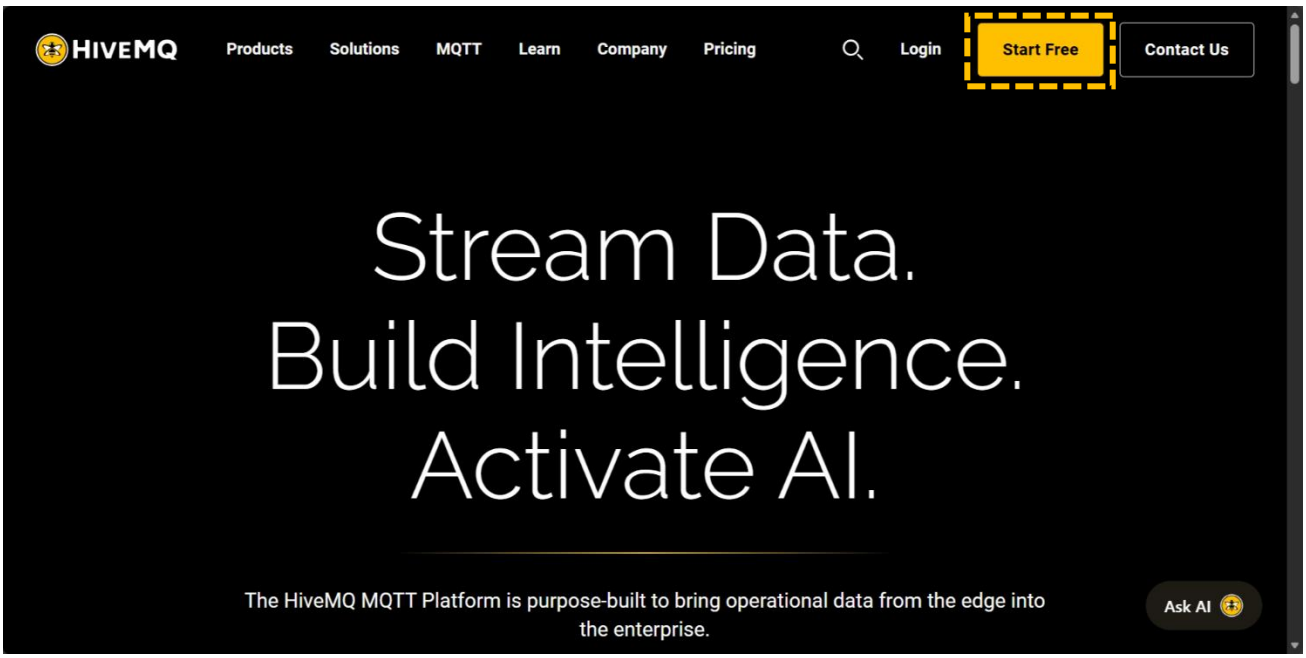
本章では MQTT ブローカーを提供している幾つかのクラウドサービスにおいて、MQTT ブローカーとそこにアクセスするためのユーザーを作成する流れを記載します。ご自身で用意した MQTT ブローカーが既にある場合や、Arsprout クラウド公式の MQTT ブローカー（提供時期未定）を利用する場合、この章はスキップしてください。

なお、ここで記載しているクラウドサービスの画面構成等は本書の記述時点のものであり、変化する可能性があります。

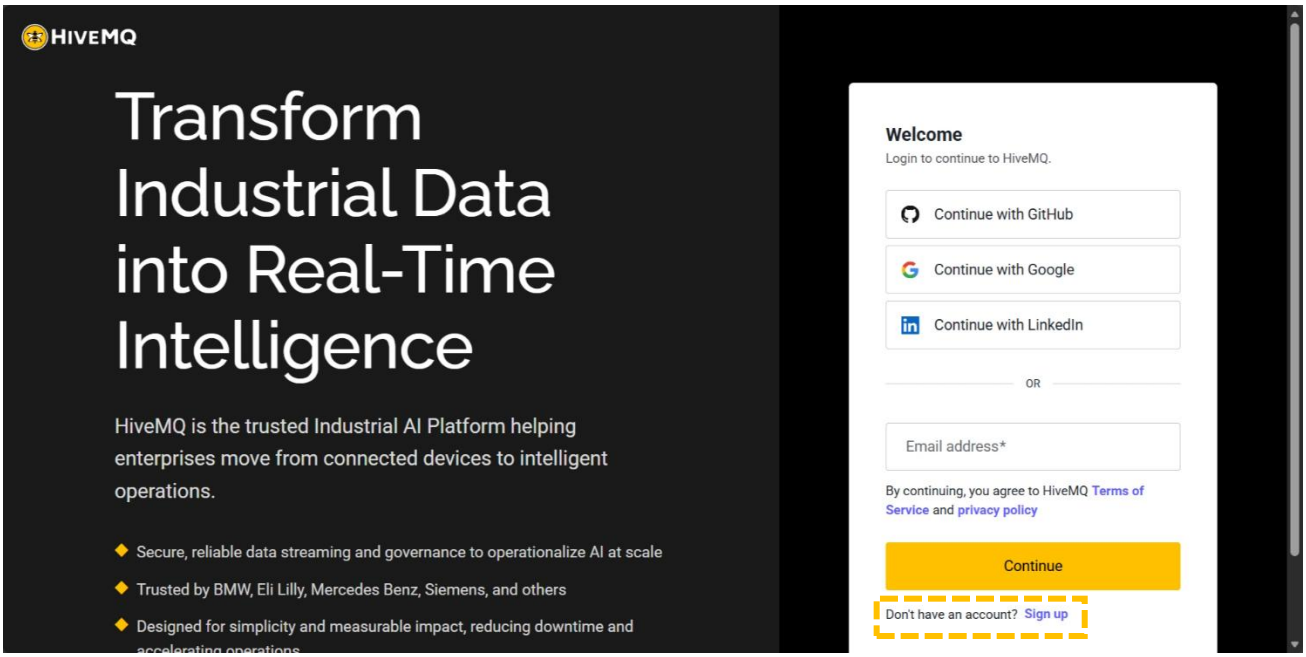
4.1. MQTT ブローカー提供サービス

4.1.1. HiveMQ

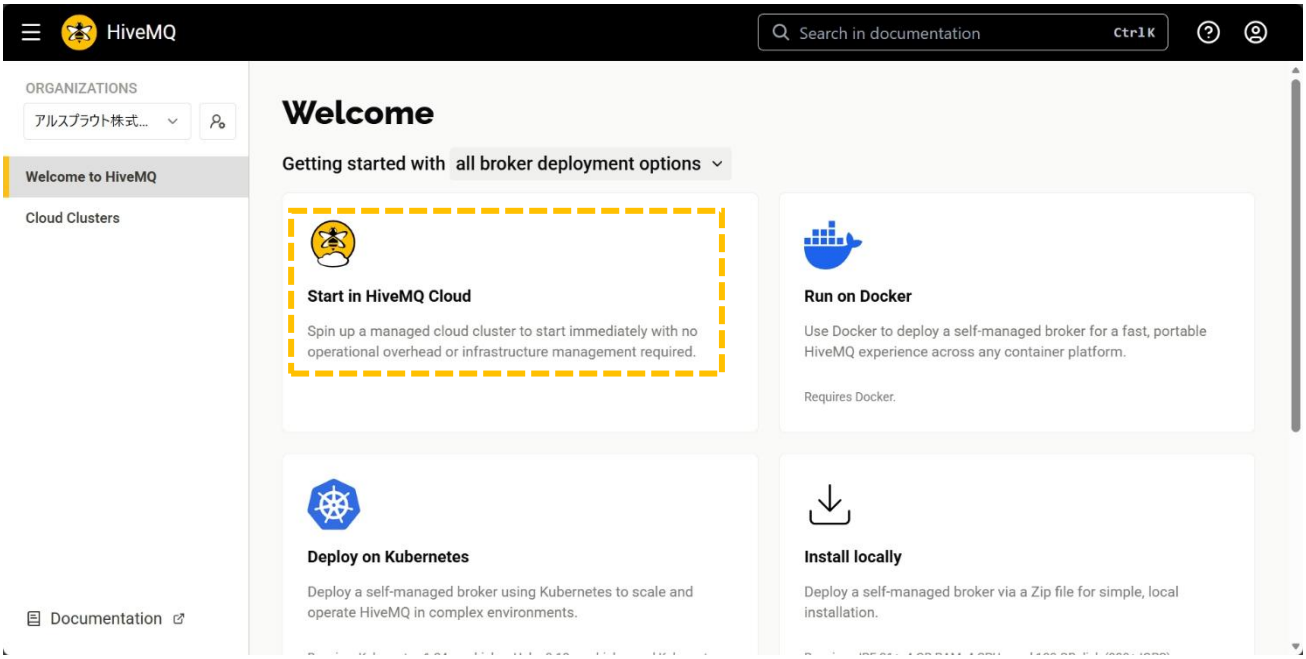
① HiveMQ（https://www.hivemq.com）にアクセスし、「Start Free」を選択します。



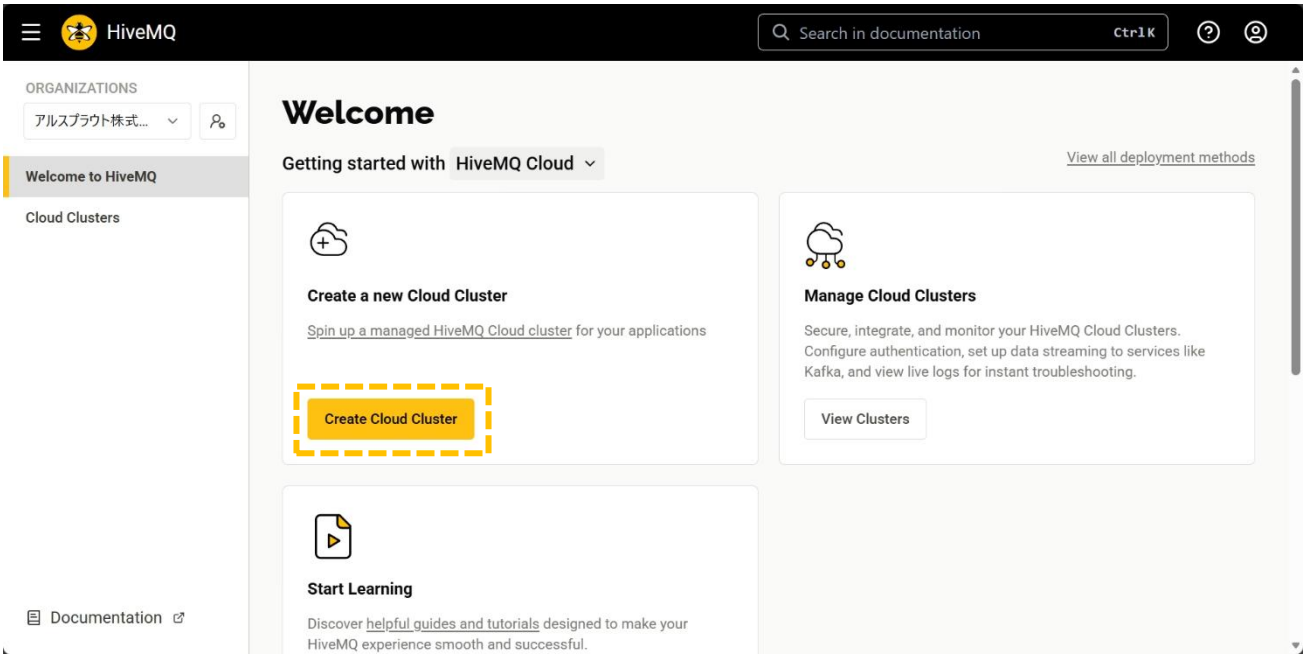
② サインアップおよびアカウント作成を行います。



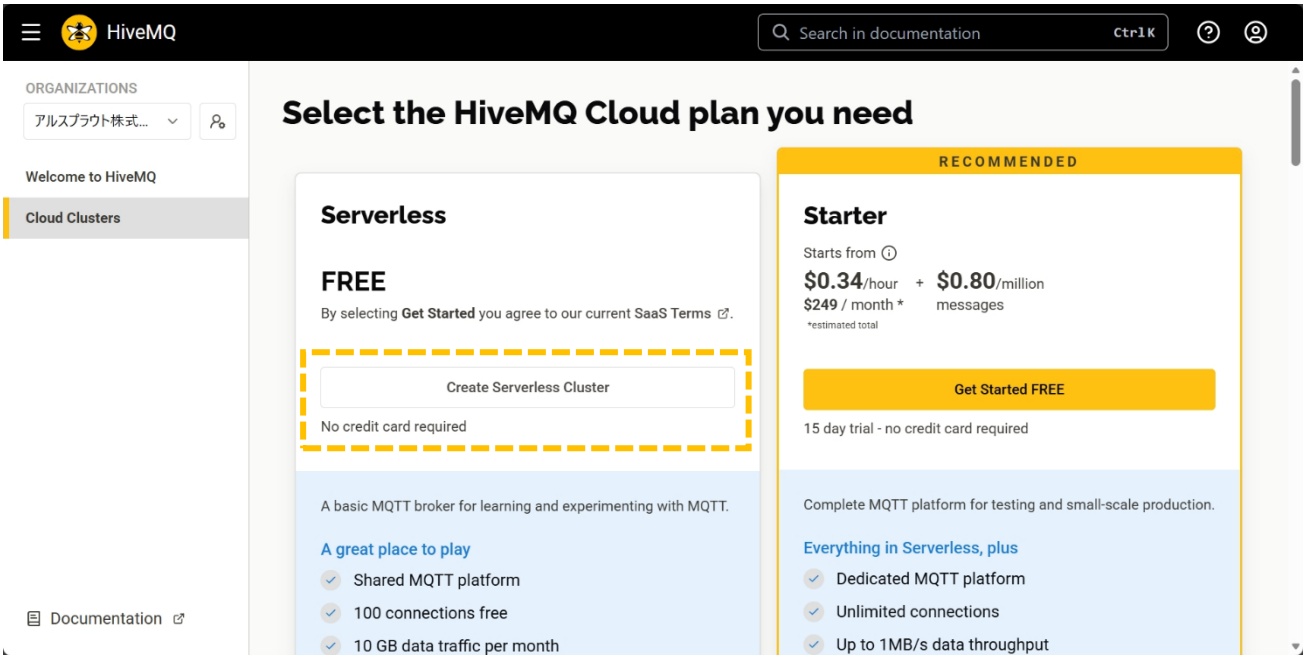
③ アカウント作成が完了し、管理コンソールが表示されたら「Start in HiveMQ Cloud」を選択します。



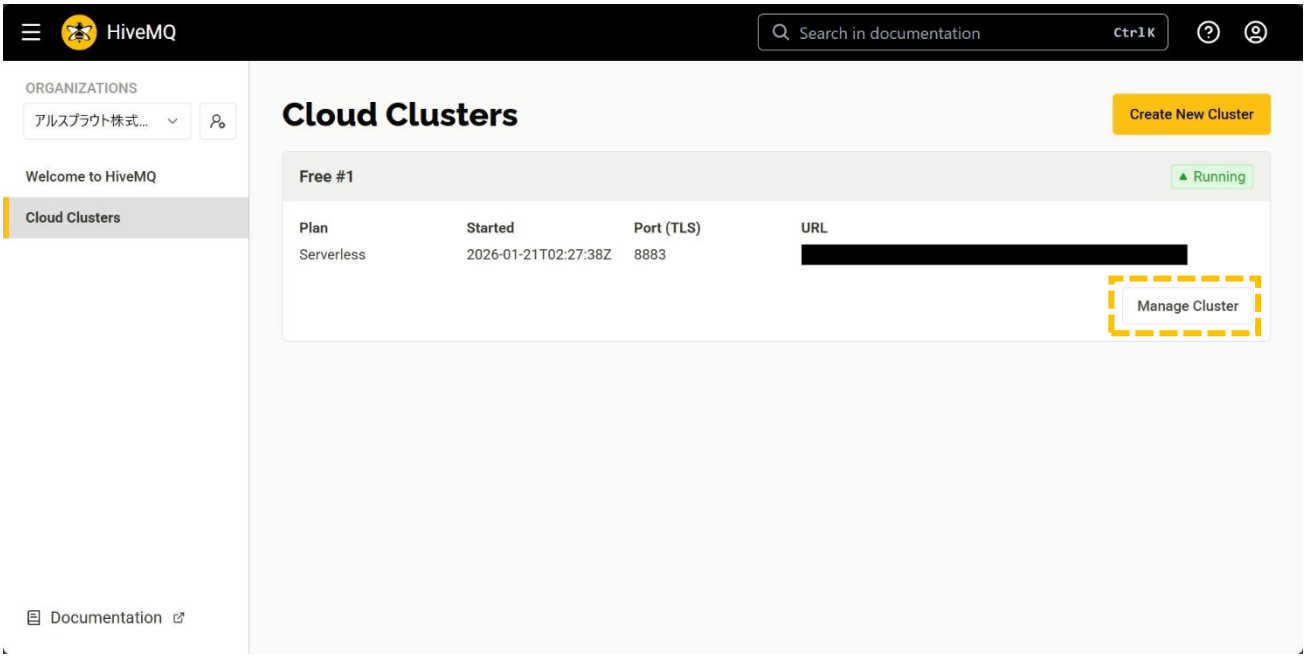
④　「Create Cloud Cluster」ボタンを押下します。



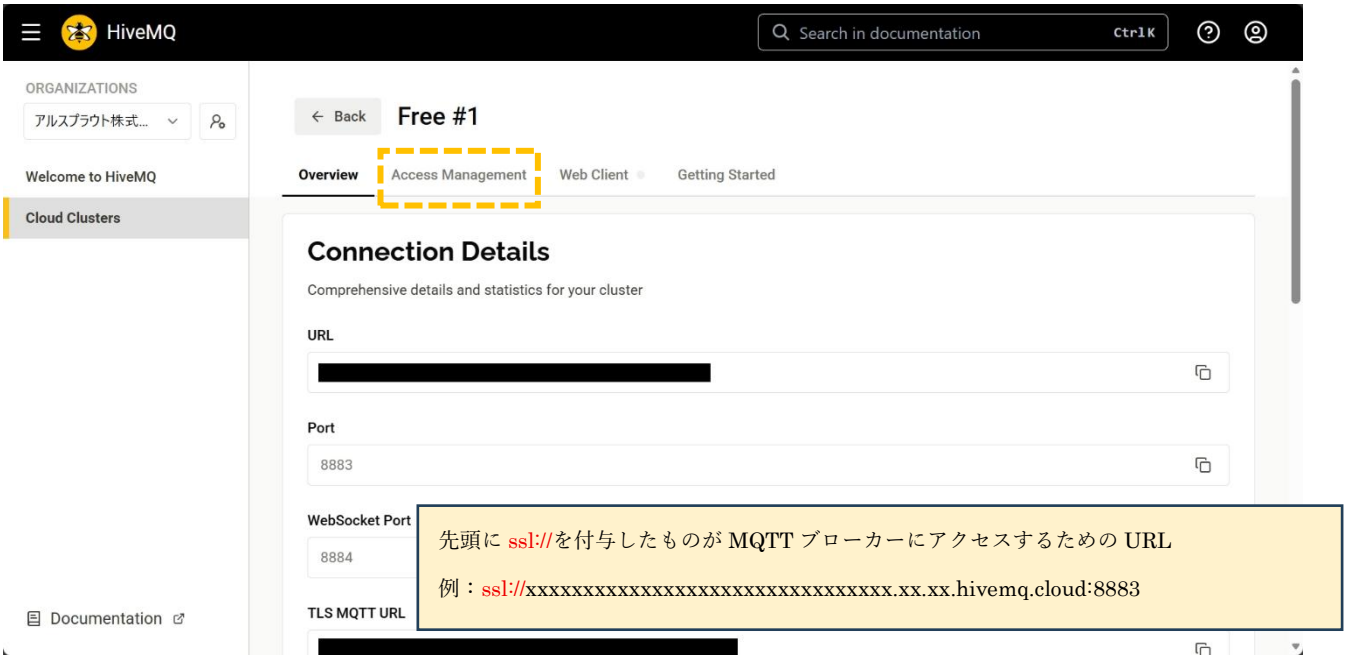
⑤　無料プランとなる「Create Servlerless Cluster」ボタンを押下します。



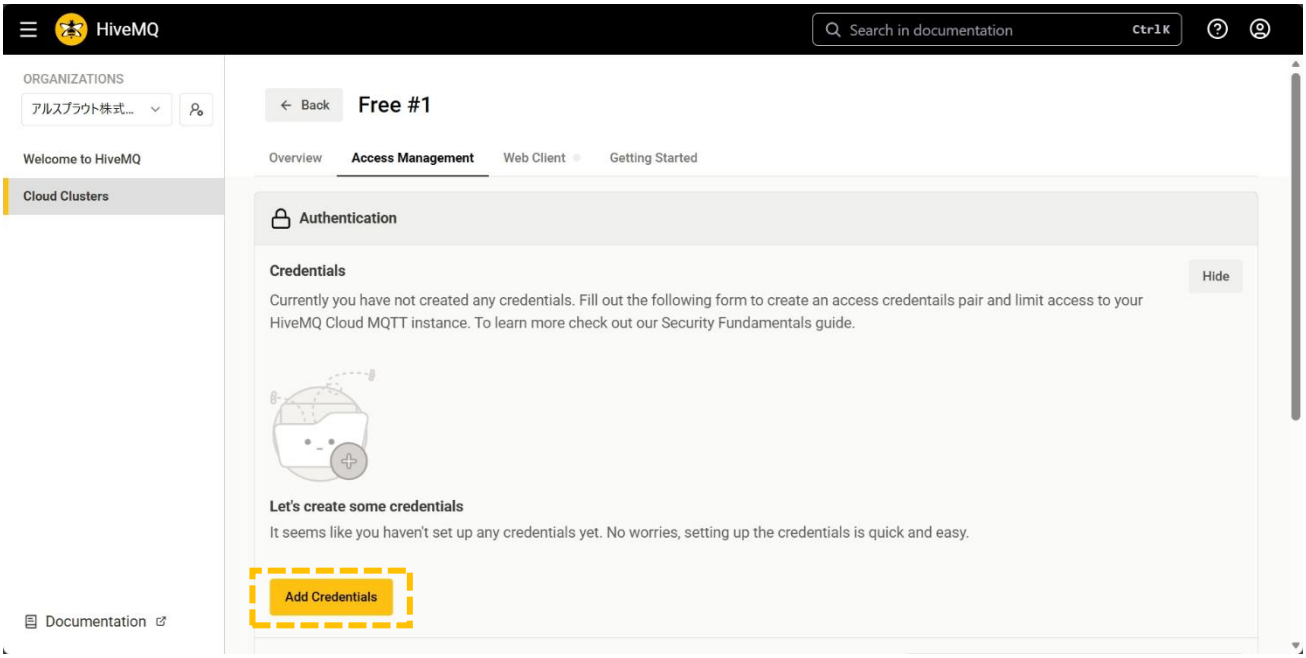
⑥　MQTT ブローカーが作成されます。「Manage Cluster」ボタンを押下します。



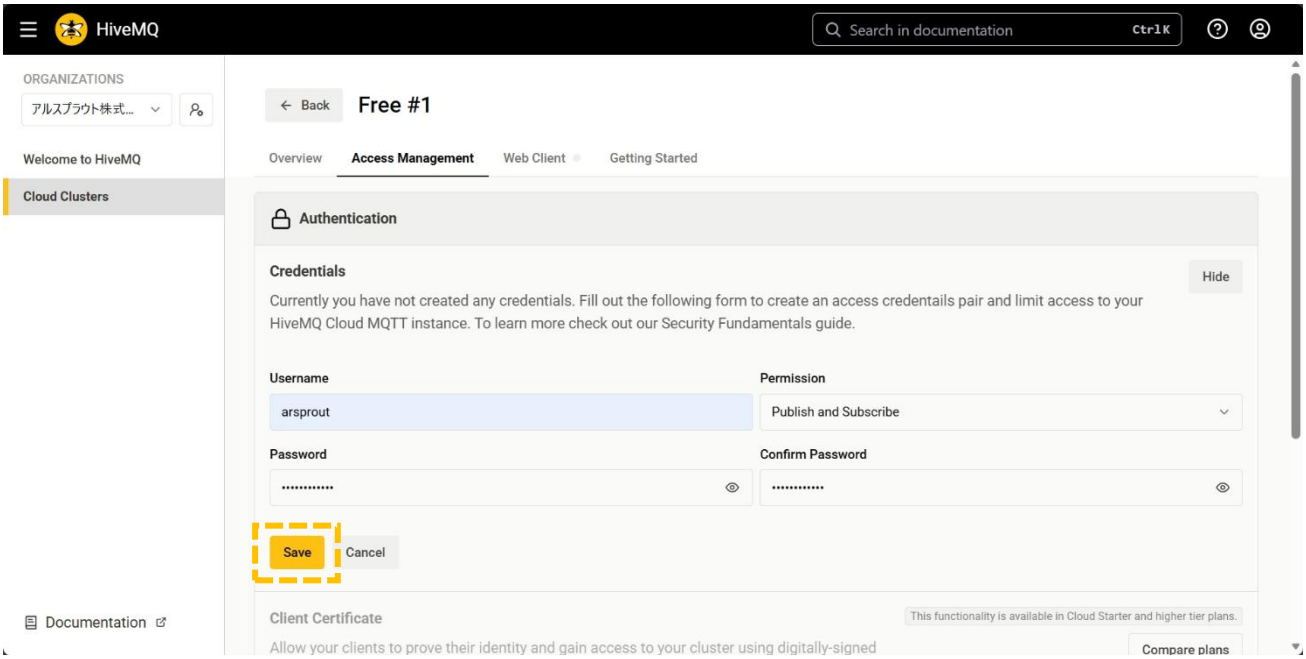
⑦　概要情報が表示されます。ここで確認できる「TLS MQTT URL」の先頭に ssl:// （全て半角文字）を付与したものが MQTT ブローカーにアクセスするための URL となります。確認ができたら「Access Management」タブを選択します。



⑧　「Add Credentials」ボタンを押下します。



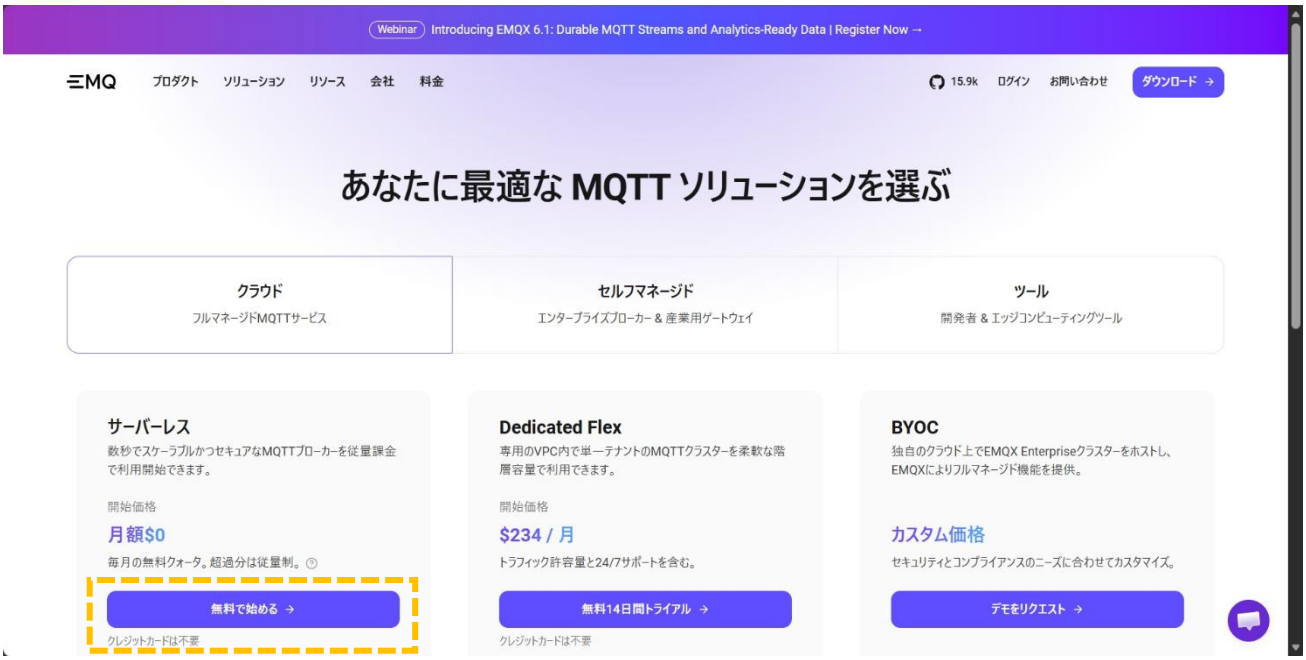
⑨　ユーザー名とパスワードを入力（パスワードは確認のため 2 箇所に入力が必要）し、Permission に「Publish and Subscribe」を選択して「Save」ボタンを押下します。これにより MQTT ブローカーに接続するためのユーザーが作成されます。



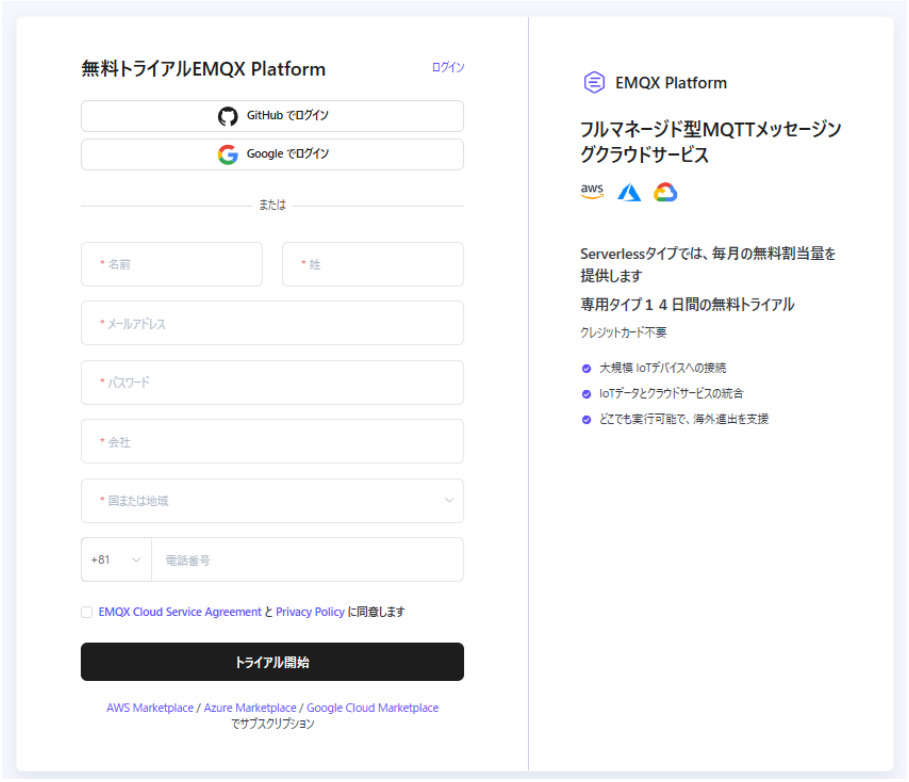
4.1.2. EMQX

① EMQX（https://www.emqx.com/ja）にアクセスし、「無料で開始」を選択します。

② 「無料で始める」を選択します。



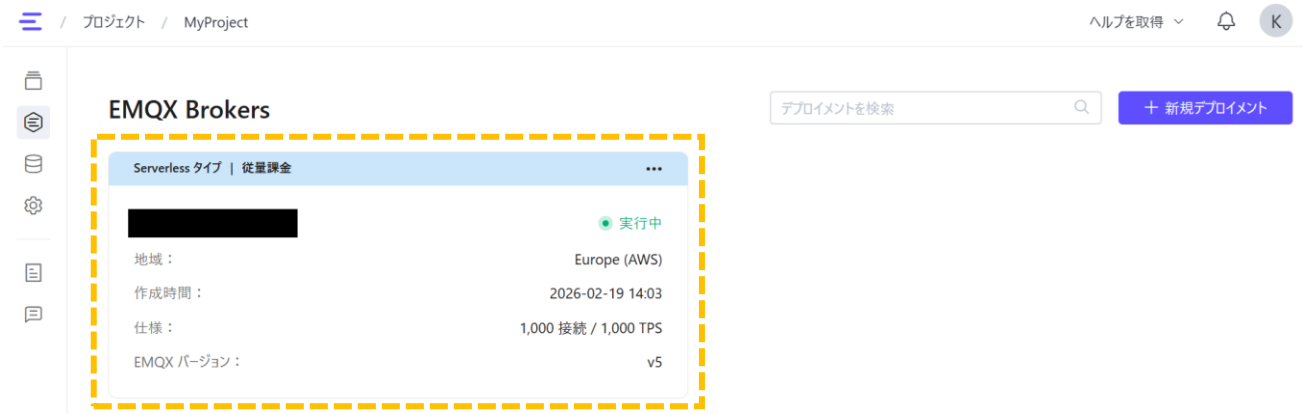
③　サインアップおよびアカウント作成を行います。



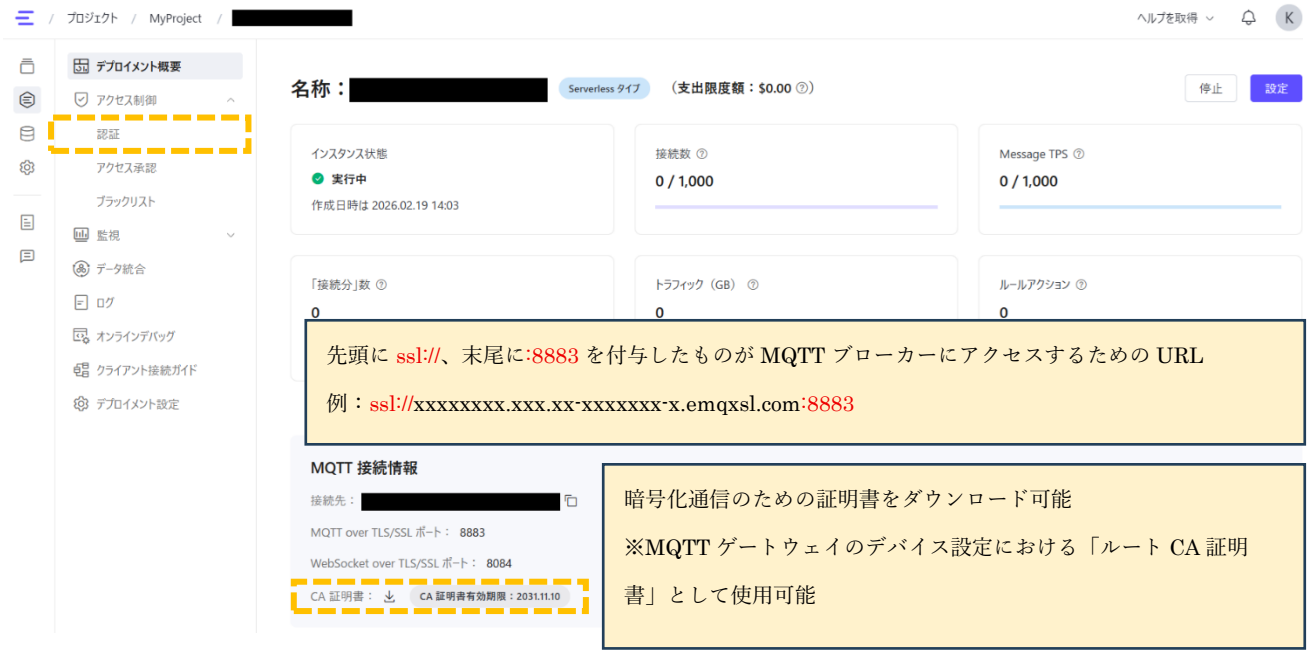
④　アカウント作成が完了するとプロジェクト名の入力を求められるため、任意の名称を入力します。ここでは「MyProject」としています。プロジェクト概要が表示されたら「＋新規 EMQX Broker」ボタンを押下します。

⑤　「Servlerless タイプ」を選択して「今すぐ導入」ボタンを押下します。

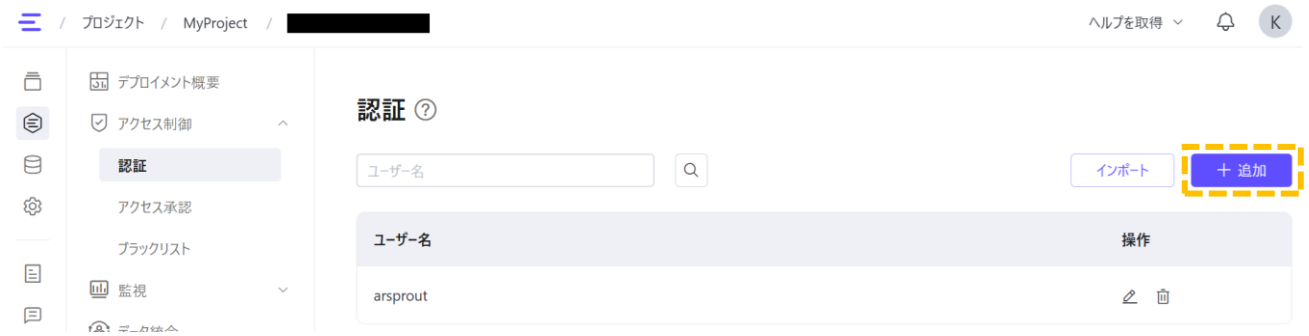
⑥　MQTT ブローカーが作成されますので選択します。



⑦ 概要情報が表示されます。ここで確認できる「接続先」の先頭に ssl://、末尾に:8883（全て半角文字）を付与したものが MQTT ブローカーにアクセスするための URL となります。また、ここでは MQTT ブローカーの証明書をダウンロードできます。確認ができたらメニューの「認証」を選択します。



⑧ 「＋追加」ボタンを押下し、ユーザー名とパスワードを入力します。これにより MQTT ブローカーに接続するためのユーザーが作成されます。

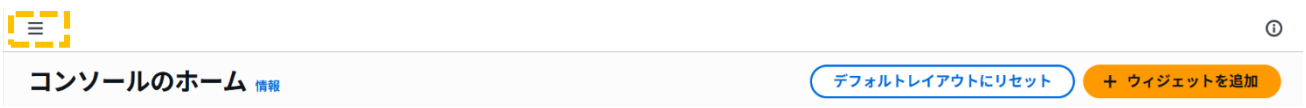


4.1.3. AWS

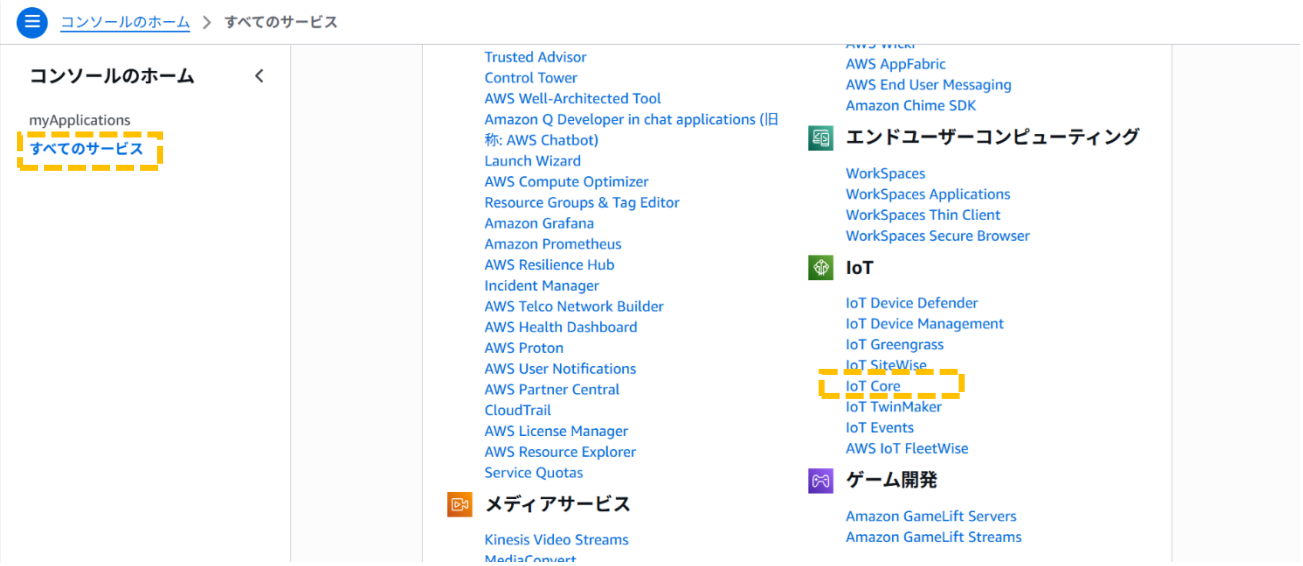
① AWS（https://aws.amazon.com/jp/console/）にアクセスし、「アカウントの作成」を選択します。

② サインアップおよびアカウント作成を行います。

③ アカウント作成が完了し、管理コンソールが表示されたらメニューを選択します。



④　メニューから、すべてのサービス、IoT Core と選択します。



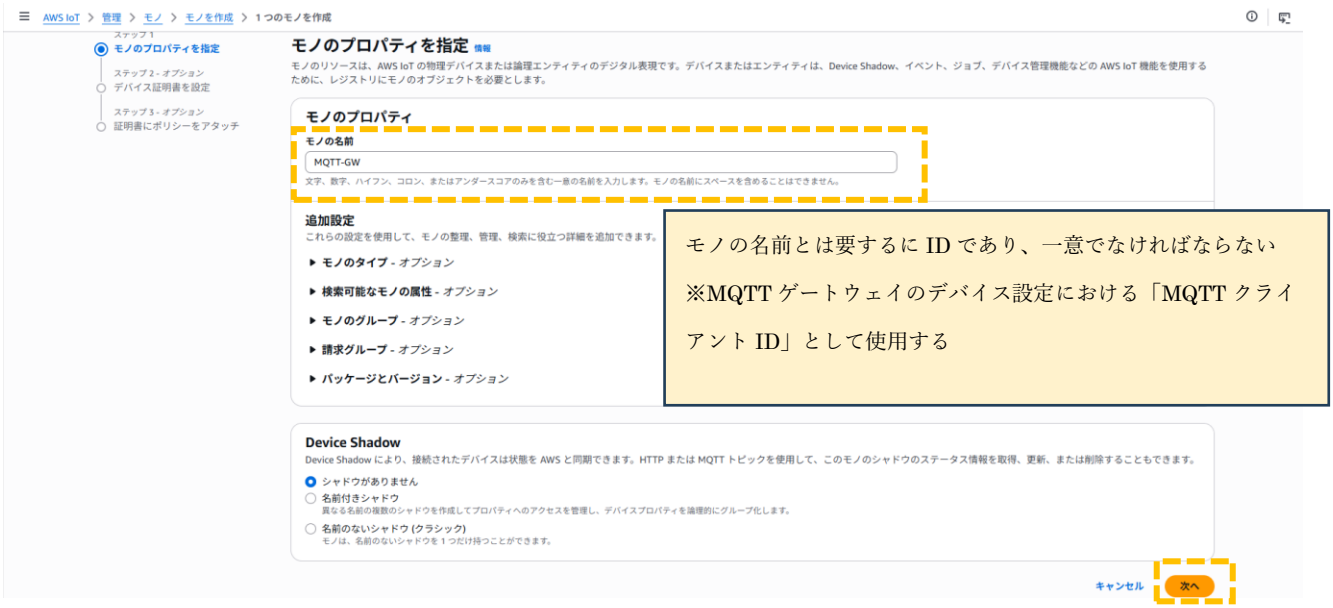
⑤　IoT Core のメニューのドメイン設定を選択します。(※既に AWS のアカウントを保有していた方へ：AWS のルートユーザーではなく IAM ユーザーを使用している場合は、ここで権限不足と表示される可能性があります。自らルートユーザーを介して IoT Core のフルアクセス権限を付与するか、アカウント情報の提供元（アカウント管理者）に IoT Core のフルアクセス権限の付与を依頼して下さい)

⑥　ドメイン設定が表示されます。ここで確認できる「ドメイン名」の先頭に ssl://、末尾に:8883（全て半角文字）を付与したものが MQTT ブローカーにアクセスするための URL となります。確認ができたらメニューの「モノ（MQTT ブローカーにアクセスするモノを定義する機能)」を選択します。

⑦　「モノを作成」ボタンを押下します。

⑧　「1 つのモノを作成」を選択して「次へ」ボタンを押下します。

⑨　任意のモノの名前を入力して「次へ」ボタンを押下します。



モノの名前とは要するに ID であり、一意でなければならない

※MQTT ゲートウェイのデバイス設定における「MQTT クライアント ID」として使用する

⑩　「新しい証明書を自動生成」を選択して「次へ」ボタンを押下します。

⑪　「ポリシーを作成」ボタンを押下します。

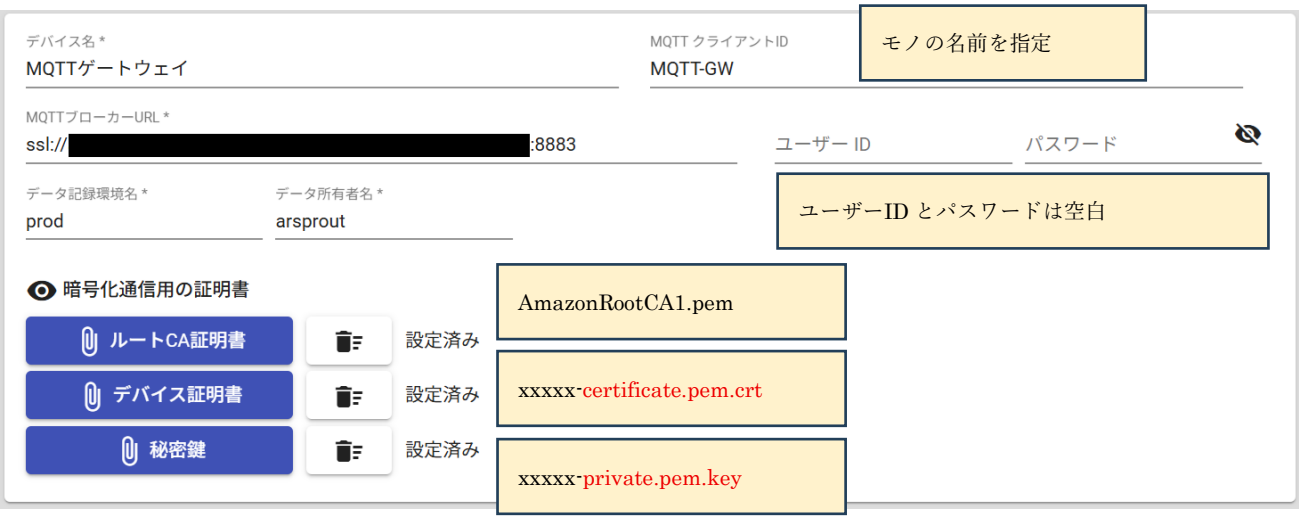
⑫　任意のポリシー名、ポリシー効果=「許可」、ポリシーアクション=「*（アスタリスク。半角文字)」、ポリシーリソース=「*（アスタリスク。半角文字)」を入力して「次へ」ボタンを押下します。

⑬　「証明書にポリシーをアタッチ（ブラウザによって別タブや別ウインドウとなります)」に戻り、作成したポリシーを選択して「モノを作成」ボタンを押下します。

⑭　証明書のダウンロードを促されますので「すべてダウンロード」ボタンを押下します。

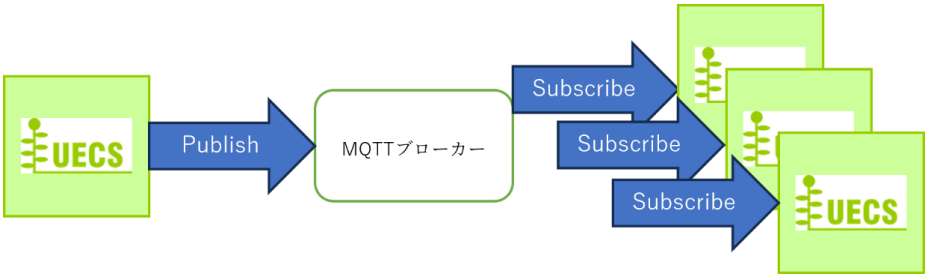
⑮ 以下のような 5 つのファイルがダウンロードできていることを確認します。ファイル名が長い等の影響で「本当に保存して良いか」のような確認が出ることがありますが、そのまま保存して下さい。

⑯ ダウンロードしたファイルは MQTT ゲートウェイのデバイス設定の暗号化通信用の証明書となります。AWS IoT Core においては、これらがユーザー認証の役割を兼ねますので、ユーザーID とパスワードは空白とします。



5. MQTT ゲートウェイの設定および運用例

本章では MQTT ゲートウェイの設定および運用について例示します。説明を簡素にするため、基本的には Publish 側、Subscribe 側ともノード一つとして記載していますが、MQTT は「Publish 側が一つに対して、Subscribe 側は多数」となる通信の実現を主目的としています。「情報を送る側が一つ、情報を受ける側が多数」の構成を前提として考えることにより、実用例をイメージしやすくなります。



5.1. 離れたノード間でセンサー値やアクチュエータ動作状態を連動させる

5.1.1. Publish（情報を送る側）側のノードの設定

室内気温センサーと天窓アクチュエータの Publish 例を記載します。

① 室内気温センサーを用意します。データソースは自ノードに接続されたデバイス(ここでは「温室 A」の室内気温)とします。CCM 識別子、部屋番号、系統番号、通し番号(room, region, order)は Subscribe 側で合わせる必要があるためメモしておきます。センサーの設定についての詳細は「Arsprout Pi ユーザーガイド[基本機能編]」を参照してください。

② 天窓アクチュエータを用意します。データソースは CCM 遠隔制御とします。CCM 識別子、部屋番号、系統番号、通し番号（room, region, order）は Subscribe 側で合わせる必要があるためメモしておきます。アクチュエータの設定についての詳細は「Arsprout Pi ユーザーガイド[基本機能編]」を参照してください。

③ 以下のような Publish 設定の MQTT ゲートウェイを用意します。

5.1.2. Subscribe 側（情報を受ける側）のノードの設定

室内気温センサーと天窓アクチュエータの Subscribe 例を記載します。

① 室内気温センサーを用意します。データソースは CCM 受信とします。CCM 識別子、部屋番号、系統番号、通し番号(room, region, order)は Publish 側に合わせます。これにより Publish 側と Subscribe 側の紐付けが行われます。

② 天窓アクチュエータを用意します。データソースはデバイス（インターロックや入出力設定は実際の制御環境に準じる）とします。CCM 識別子、部屋番号、系統番号、通し番号（room, region, order）は Publish 側に合わせます。これにより Publish 側と Subscribe 側の紐付けが行われます。

③ 以下のような Subscribe 設定の MQTT ゲートウェイを用意します。MQTT ブローカーURL、データ記録環境名を始めとする設定は Publish 側に合わせます。ただし、MQTT クライアント ID を指定する場合は、同じ MQTT クライアント ID を使いまわさないようにしてください。MQTT クライアント ID を使いまわすと、通信そのものの問題が生じる可能性があります。

5.1.3. Publish 側と Subscribe 側の連動

設定が成功していれば、Publish 側と Subscribe 側が連動するようになります。センサーについては Publish 側のセンサー値が Subscribe 側に伝搬します。アクチュエータについては Publish 側での操作が Subscribe 側に伝搬（※）します。

※但し、Subscribe 側のアクチュエータが WEB 強制操作モード以上の優先度で稼働している場合は Subscribe 結果が無視されるため伝搬しません。

6. 巻末付録

6.1. トピックおよびメッセージの仕様

MQTT ゲートウェイで Publish/Subscribe の設定を行うと、割り当てたコンポーネント/CCM に対応するトピックにて、メッセージが Publish/Subscribe されます。トピックとは MQTT においてメッセージの在り処を示す文字列です。ファイルシステムにおけるファイルパス（Windows における「C:¥Program Files」のようなもの）と同様です。

ご自身で管理している MQTT ブローカーを使用している場合、これらの内容は MQTT ブローカーの管理コンソール等で実際に確認することができます。

トピックの仕様
data/gw/データ記録環境名/データ所有者名/部屋番号/系統番号/通し番号/CCM 識別子 ※データ記録環境名、データ所有者名は MQTT ゲートウェイのデバイス設定の値です ※部屋番号、系統番号、通し番号、CCM 識別子は UECS の仕様に基づく値です 【トピックの例】 data/gw/prod/arsprout/1/1/1/InAirTemp
メッセージの仕様
{“head”: {“timestamp”: UNIX タイムスタンプ値}, “body”: {“value”: UECS の CCM 値, “priority”: UECS の優先順位}} ※JSON 形式で記述されたメッセージです 【メッセージの例】 {“head”: {“timestamp”: 1577804400}, “body”: {“value”: 18.0, “priority”: 30}}

表 4：トピックおよびメッセージの仕様