UDC 004.273

COMPLEX SELECTION CRITERIA OF CLOUD GATEWAY

Kurdecha V.V., Ishchenko I.O., Zakharchuk A.G.
Institute of Telecommunication Systems,
National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute"
E-mail: ivanishchenkoo@gmail.com

КОМПЛЕКСНИЙ КРИТЕРІЙ ВИБОРУ ХМАРНОГО ШЛЮЗУ

У даній статті описуються дві реалізації хмарних шлюзів, що надаються компаніями Microsoft і Amazon. Клієнти хочуть набагато більше, ніж збір даних з Інтернету кінця в кінець речей платформи. Microsoft і AWS містять різні компоненти для збору, зберігання і аналізу даних.

This article describes two implementations of the Cloud Gateways, provided by Microsoft and Amazon companies. Customers want much more than data collection from an end-to-end Internet of Things platform. Microsoft and AWS platforms contain different components for collecting, storing and analyzing data.

The Cloud Gateway is a very important part of every IoT system, because it handles a huge number of devices at scale and a lot of incoming messages per second into the system [1]. For this reason, big companies started to develop their solutions to provide such a gateway and simplify the ingestion and communication part of an IoT solution. Microsoft announced its Azure IoT Hub and Amazon replied with its AWS IoT platform. In this issue two Cloud Gateway implementations of Azure IoT Hub and AWS IoT platform were compared.

IoT Hub is the new entry in the Microsoft Azure offer; it's a service that enables bi-directional communication between devices and our business engine in the cloud. The communication channel is reliable and secure and the authentication is per-device using credentials and access control [2].

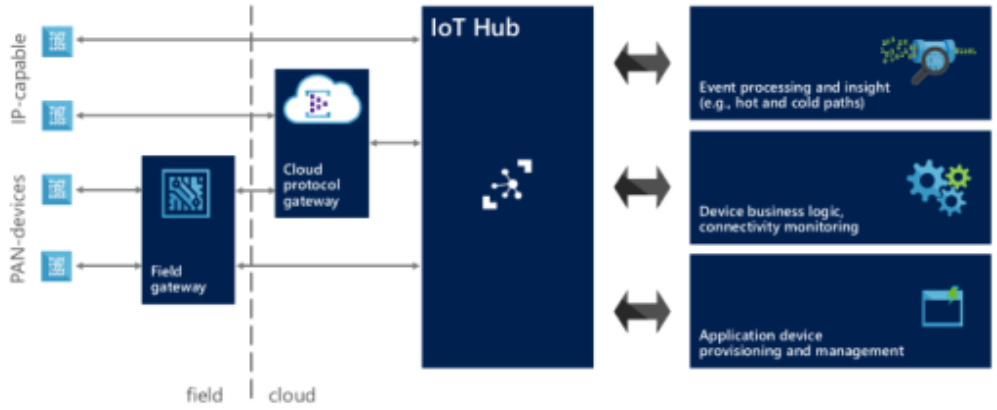

Fig. 1. IoT architecture with IoT Hub.

IoT Hub has an identity registry where it stores all information about provisioned devices. This information is not related to devices metadata (they are up to you in your IoT solution that uses IoT Hub, for example manufacturer and firmware/software version info) but it is related to identity and authentication. It provides monitoring information like connection status (connected/disconnected) and last activity time; you are also able to enable and disable the devices using this registry. Of course, IoT Hub exposes another endpoint (device identity management) to create, retrieve, update and delete devices.

AWS IoT has the same objectives as IoT Hub but reaches them in a different way.

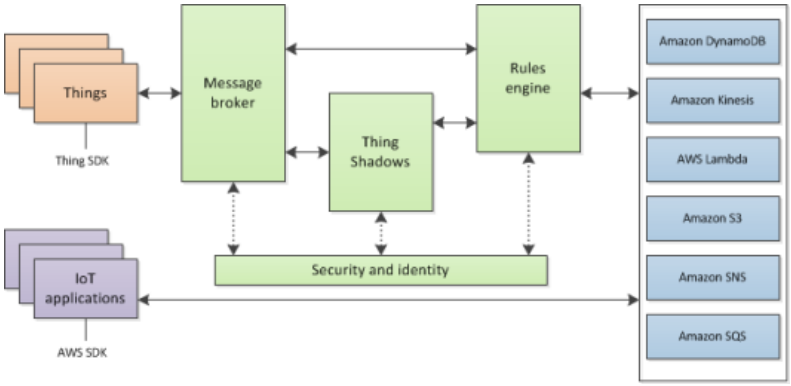

Fig. 2. AWS IoT data services

AWS IoT provides the telemetry data to the system publishing states from devices to cloud; the platform provides the command pattern changing the desired state for a thing shadow (of course changing the state for a device means to request an action). AWS IoT has a thing registry that contains devices related to information and allows adding custom attributes that are part of the devices metadata (for example manufacturer, serial number) [3]. The interaction with the thing registry to create, delete and update things is enabled with the AWS CLI (Command Line Interface) that provides all such operations.

The connection established between devices and IoT Hub is TLS (Transport Layer Security) based so that the communication is encrypted to guarantee data confidentiality; the server is authenticated thanks to its own X.509 certificate sent to the device during the TLS handshaking. The authentication is provided by IoT Hub verifying a token (sent from the device) against the shared access policies and device identity registry security credentials.

AWS IoT relies on TLS protocol so that the communication with the message broker is encrypted and the client is authenticated using the mutual authentication (so with X.509 client certificate too). The certificates can be created, activated and revoked using the AWS CLI or the AWS online console; of course we can also use a certificate that is already in our possession [4].

It is possible to support additional protocols using the Azure IoT Protocol Gateway framework that provides protocol adaptation from a different protocol

(MQTT or any other custom protocol) to AMQP with direct access to the IoT Hub. MQTT is the official supported protocol for AWS IoT that the message broker uses for publishing and subscribing messages on topics.

The IoT Hub price is defined starting from the IoT Hub unit concept related to the maximum number of device connected and the number of messages transmitted per day. Each unit allows handling up to 500 devices and we can allocate a maximum number of 200 units. Of course, the IoT Hub provides operation throttles for all types of operation like identity registry operations, device connection and finally device to cloud and cloud to device operations. To try out the IoT Hub for free, there is a free tier which enables to connect up to 10 devices with a maximum of 3,000 messages per day (from all devices). Pay attention that current price reflects a preview discount of 50%. The AWS IoT pricing is based on million messages exchanged quota. The total cost is evaluated both on publishing and delivering messages from/to devices and applications. A message is a 512-byte block of data processed by AWS IoT. To start trying the platform we can create a account and use the free tier which gets you started with 250,000 free messages (published or delivered) per month, for 12 months without limitations on the number of connected devices.

Conclusion. It is interesting to see how each provider is tackling the IoT problem space. Microsoft and Amazon developed their platforms with different choices starting from the underlying protocols used for communication : AMQP vs MQTT. Microsoft has already used AMQP for all services under Service Bus umbrella so it seems to be a logical choice for IoT Hub.

Pricing is completely different and it could be one reason for your choice depending on your IoT business.

References

1. Globa L.S., Kurdecha V.V., Ishchenko I.O., Zakharchuk A.G. An approach to the Internet of Things system architecture. CADSM'2017.
2. R. Want, B. N. Schilit, and S. Jenson “Enabling the internet of things” Computer, vol. 48, no. 1, pp. 28–35, Jan 2015.
3. Hongki Cha, Wonsuk Lee, Jonghong Jeon “Standardization strategy for the Internet of wearable things”, Republic of Korea, ICTC 2015.
4. Jorge E. Luzuriaga , Miguel Perez , Pablo Boronat , Juan Carlos Cano, Carlos Calafate , Pietro Manzoni “A comparative evaluation of AMQP and MQTT protocols over unstable and mobile networks”, IEEE 12th Consumer Communications and Networking Conference, 2015.