



Het ESP RainMaker-verhaal

hoe we “jouw” IoT-cloud bouwden

Amey Inamdar, Espressif

Met de cloud verbonden apparaten zijn een belangrijk onderdeel van het IoT. Echter, met cloudproviders die een volledige oplossing bieden, verliezen de fabrikanten van die apparaten controle over de data. Het bouwen van een eigen cloudoplossing vereist daarentegen een aanzienlijke inspanning. Bij Espressif zagen we deze problemen en besloten we een oplossing te bieden die het beste van beide werelden biedt.

De ‘I’ in IoT staat voor internet. Het internet - en daarmee ook de cloud - is een integraal onderdeel van communicerende objecten om de voordelen van connectiviteit te benutten. Apparaten die de cloud gebruiken om gegevens te verzenden en opdrachten te ontvangen, bieden niet alleen toegevoegde waarde aan eindgebruikers, maar ook aan hun fabrikanten. Het hebben van een eigen IoT-cloud brengt echter beveiligings- en privacyoverwegingen met zich mee voor de consument en technische uitdagingen voor de eigenaar. Het was een logische keuze voor veel fabrikanten om in zee te gaan

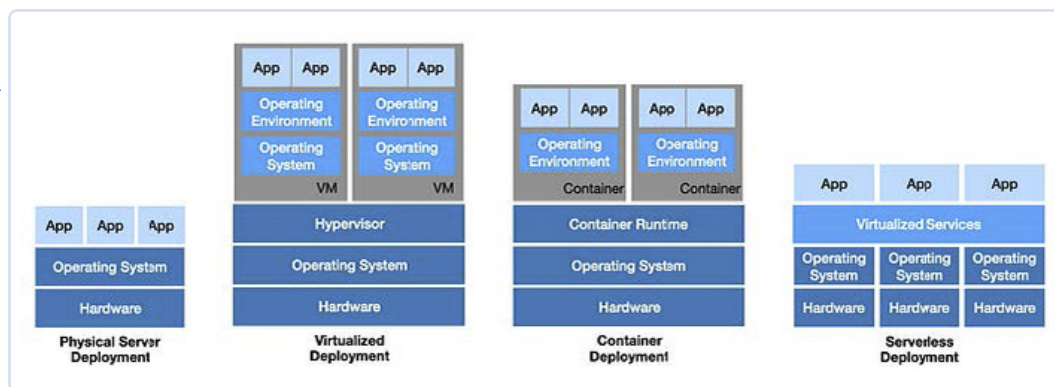
met leveranciers van volledige IoT-oplossingen die cloud-, embedded- en mobiele applicaties boden. Ze kwamen er echter al snel achter dat deze oplossing onvoldoende differentiatiemogelijkheden bood en niet de vereiste controle over de data gaf. Sommige fabrikanten die de mogelijkheid hadden besloten hun eigen cloud te bouwen en te onderhouden, maar dit vereist een aanzienlijke inspanning.

Bij Espressif zagen we deze problemen en besloten we een oplossing te bieden die het beste van beide werelden biedt. Aan de ene kant wilden we een cloud met alles erop en eraan, en aan de andere kant wilden we onze klanten volledig eigendom, controle en aanpasbaarheid bieden, zonder dat ze vanaf niets hoefden te beginnen. Dit was de aanzet tot ESP RainMaker.

Kies de juiste architectuur

De eerste keuze die we moesten maken was welke architectuur we wilden gebruiken (**figuur 1**).

Je hebt misschien wel eens horen zeggen dat de cloud gewoon de computer van iemand anders is. Dit is inderdaad zo. Maar waar het om gaat, is hoe die computer wordt gebruikt om je applicatie te hosten. Met de opkomst van virtualisatietechnologieën worden krachtige servers gebruikt om meerdere applicaties volledig geïsoleerd van elkaar te hosten op dezelfde hardware. Met virtuele machines (denk aan VMWare) kun je meerdere instanties van besturingssystemen op dezelfde hardware draaien, terwijl je met containertechnologie (Kubernetes, Docker) parallelle besturingsomgevingen op dezelfde hardware kunt hebben.



Figuur 1. Evolutie van cloudarchitecturen.

Dit klinkt allemaal heel mooi, maar je moet wel nadenken over welke software je moet draaien en onderhouden en hoe je moet schalen met variërende belastingen (dit is de taak van zogenaamde DevOps-engineers). Toen we besloten onze klanten hun eigen IoT-cloud te geven, wilden we ze niet belasten met deze taken. Dat is waar de relatief nieuwe techniek van de cloud-zonder-server uitkomst bood met mogelijkheden voor het bouwen van een eenvoudig te onderhouden IoT-cloud.

Een architectuur zonder servers maakt servers niet overbodig. De abstracties worden gewoon op een hoger niveau aangeboden. Je krijgt bijvoorbeeld een MQTT-broker zonder dat je je zorgen hoeft te maken over welke MQTT-software hij gebruikt, hoeveel verbindingen hij aankan of op welk platform hij draait. Amazon Web Services (AWS) levert zulke diensten, waardoor we onze droom IoT-cloud konden implementeren. Van de vele beschikbare diensten biedt AWS IoT Core een MQTT-broker, DynamoDB biedt een NoSQL database, S3 biedt opslag en, belangrijk, Lambda biedt een "function-as-a-service", waarbij je algoritmes kunt implementeren zonder je zorgen te maken over waar deze moeten worden uitgevoerd.

Met de op AWS managed services gebaseerde implementatie lijkt ESP RainMaker op een combinatie van het configureren van verschillende services en hun interactie, en het gebruik van Lambda-functies. Belangrijk is dat het geheel past in elk AWS-account. Dit geeft ons de mogelijkheid om een IoT-cloud te maken die klanten in hun eigen AWS-account kunnen implementeren, zodat ze volledige controle hebben over de data en het systeem.

Schaalbaarheid

Een IoT-cloud heeft vaak veel objecten en gebruikers (via mobiele apps of andere clients zoals spraakassistenten) die verbinding maken met de cloud en berichten uitwisselen. Afhankelijk van de branche varieert dit aantal. Voor fabrikanten van consumentenapparatuur kunnen de aantallen in de miljoenen lopen. Daarom is het belangrijk om ervoor te zorgen dat de cloud kan worden opgeschaald naar grote aantallen.

Met AWS managed services mag je aannemen dat je je geen zorgen hoeft te maken over de schaalbaarheid van de cloud. Hoewel dit groten-deels waar is, is het ook zo dat een service bepaalde limieten kan hebben. Het maximale aantal Lambda-functies dat gelijktijdig kan worden uitgevoerd in een AWS-account is bijvoorbeeld beperkt. Dit vereist een zorgvuldig ontwerp om ervoor te zorgen dat het systeem zoveel mogelijk berichten goed kan afhandelen dankzij een adequaat wachtrijsysteem. Het vereist ook dat objecten en mobiele apps op een goede manier met problemen kunnen omgaan.

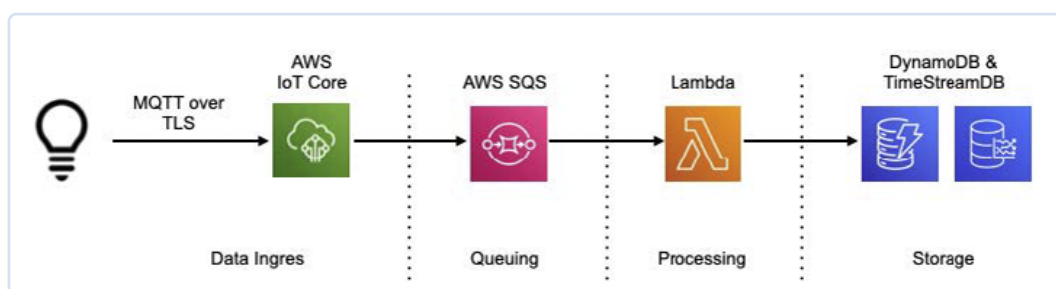
ESP RainMaker kan momenteel meer dan 3,5 miljoen objecten en gebruikers aan die verbinding maken en communiceren via de cloud (figuur 2).

Operationele kosten

In een traditionele cloudarchitectuur betaal je meestal voor computing, opslag en geheugen voor je virtuele machines. In de serverloze architectuur verandert de eenheid van facturering. Je moet betalen voor de daadwerkelijk gebruikte resources, zoals het aantal verbonden objecten en de totale verbindingstijd, het aantal MQTT-berichten, het aantal keer dat de database gelezen/geschreven is, de hoeveelheid gegevens die opgeslagen is in S3 Storage en de resources die verbruikt zijn door uitgevoerde Lambda-functies. Hoewel dit een pay-as-you-go prijs is, kan het veel meer kosten als je architectuur en implementatie niet geoptimaliseerd zijn.

De operationele kosten voor de klanten waren een van de belangrijkste criteria bij het ontwerp van ESP RainMaker. Het doel was gemakkelijk te bepalen. We wilden een cloud backend ontwikkelen dat ook zou werken voor een slimme gloeilamp – waarschijnlijk het goedkoopste IoT-object. Het heeft veel moeite gekost om de AWS-services zo te kiezen dat ESP RainMaker dit doel kon bereiken. Verder is het flink getuned om het aantal database lees/schrijfbewerkingen te optimaliseren; de implementatietaal moest zo weinig mogelijk resources gebruiken, enzovoort.

ESP RainMaker heeft momenteel meerdere klanten die daadwerkelijk goedkope IoT-lampen verkopen die voldoen aan de operationele kostenvereisten.



Figuur 2. Een voorbeeld van verwerking van objectberichten in ESP Rainmaker.

Figuur 3. ESP RainMaker componenten.



Gegevensbeveiliging en privacy

Het gebruik van AWS managed services hielp ook bij het waarborgen van databeveiliging. Ze bieden standaardmethoden voor authenticatie, autorisatie, versleutelde data-opslag, firewalls enzovoort. De AWS IoT Core MQTT-broker biedt bijvoorbeeld objectverificatie op basis van X.509-certificaten, waardoor een apparaat de cloud kan verifiëren en de cloud een apparaat. IAM-beleidsregels zorgen voor fijnmazige autorisatiecontrole. Web Application Firewall (WAF) biedt bescherming tegen DoS- en DDoS-aanvallen op de RESTful service-eindpunten.

ESP RainMaker gebruikt al deze beveiligingsmethoden op de voorgeschreven manier en zorgt voor de juiste authenticatie en autorisatie met behulp van standaardmethoden; er is geen onnodige privilege-escalatie. We hebben ook samengewerkt met een test- en certificeringsbedrijf om ervoor te zorgen dat ESP RainMaker voldoet aan de vereisten voor dataprivacy. Hierdoor is het voor klanten eenvoudig om te voldoen aan GDPR-compliance wanneer ze ESP RainMaker inzetten in hun eigen account.

De cloud alleen is niet genoeg

Hoewel de cloud een belangrijk onderdeel is van de complete IoT-oplossing, is het niet genoeg. Gezien vanuit het verbonden IoT-object is het, naast het hebben van de cloud, net zo belangrijk om over goede firmware te beschikken. Ook zijn er apps nodig voor de configuratie en de bediening. En dan is er nog de integratie met spraakassistenten, die de meest natuurlijke manier zijn geworden voor gebruikers om te communiceren met verbonden apparaten in het slimme huis (**figuur 3**). Bij Espressif hebben we al deze componenten gemaakt. De ESP RainMaker SDK is volledig open source en eenvoudig te integreren met behulp van de vele voorbeelden. De iOS en Android voorbeeldapps zijn ook volledig open source. ESP RainMaker ondersteunt ook Alexa en Google Assistant skills.

De cloud moet ook gebruikt kunnen worden voor het verbeteren van een product wegens voortschrijdend technisch- en zakelijk inzicht. Hiervoor hebben we ESP-Insights gebouwd die remote device logs, crash analytics, metrics monitoring en, niet te vergeten, uitgebreide dataqueries mogelijk maakt. Dit is allemaal beschikbaar via het ESP RainMaker webdashboard, dat ook OTA-updates, objectgroepering, rolgebaseerde toegangscontrole en bedrijfsinzichten biedt.

Vorbereiden op de toekomst

Het Matter protocol is bedoeld voor de broodnodige standaardisatie voor smart home apparaten. Matter is een lokaal netwerkprotocol dat zich niet speciaal met de cloud bemoeit. Toch speelt de cloud een rol bij objectbeheer en de toegang op afstand tot de objecten. ESP RainMaker ondersteunt Matter zodat je je eigen Matter ecosysteem kunt creëren. Een combinatie van ESP RainMaker cloud backend en telefoon apps implementeren Matter, inclusief de complete PKI-infrastructuur die hiervoor nodig is. De databases voor object, gebruiker en ACL zijn veilig gesynchroniseerd over meerdere apps. Als je een

smart home controller hebt die verbinding maakt met ESP RainMaker, kunnen je mobiele apps niet alleen jouw apparatuur op afstand bedienen, maar ook alle andere Matter objecten.

We hebben ook ondersteuning aangekondigd voor de integratie van Mesh-Lite netwerktopologie met ESP RainMaker. Mesh-Lite is een speciaal WiFi mesh netwerk voor IoT-apparaten om een groter bereik en dode hoeken in huis te dekken. Op de ESP32 familie gebaseerde IoT-objecten vormen toegangspunten om verbinding te maken met andere objecten en vormen zo een mesh-topologie. Met ESP RainMaker kunnen gebruikers eenvoudig een mesh-netwerk opzetten en de nodes erin hangen.

Aan de slag

We hebben de ESP RainMaker-cloud gebouwd met overwegingen die elke fabrikant zou maken bij het opzetten van een eigen IoT-cloudplatform. Het belangrijkste is dat Espressif dit aanbiedt als een volledig rebrandable en aanpasbaar platform. Voor ontwikkelaars bieden we ook een door Espressif beheerde ESP RainMaker instance, samen met apps (beschikbaar in de diverse app stores). Je kunt het webdashboard gebruiken voor objectbeheer en OTA firmware upgrades uitvoeren. Hiermee kun je niet alleen de functionaliteit evalueren, maar ook je projecten doen voor persoonlijk gebruik.

Om aan de slag te gaan, kun je een van de Espressif-ontwikkelboards gebruiken en de stappen in de Get Started guide volgen [1].

230621-03



Over de auteur

Amey Indamar is hoofd technische marketing bij Espressif. Hij heeft 20 jaar ervaring op het gebied van embedded systemen en communicerende objecten, met functies in engineering, productmanagement en technische marketing. Hij heeft met veel klanten samengewerkt om succesvolle communicerende apparaten te bouwen op basis van WiFi- en Bluetooth-connectiviteit.

WEBLINK

[1] Aan de slag met ESP Rainmaker:
<https://rainmaker.espressif.com/docs/get-started>