Alma Mater Studiorum
Università di Bologna
Campus di Cesena

DIPARTIMENTO DI INFORMATICA – SCIENZA E INGEGNERIA
Corso di Laurea in Ingegneria e Scienze Informatiche

IL PROTOCOLLO MATTER:

LA RIVOLUZIONE OPEN SOURCE NEL CAMPO DELL'IOT

Elaborato in:

Reti di Telecomunicazioni

Relatore:
Prof. Franco Callegati

Presentata da:
Alessandro Palladino

Sessione IV
Anno Accademico 2021-2022

"Memento Audere Semper"
G. D'Annunzio

Sommario

Nel mondo di oggi l'Internet of Things sta assumendo sempre più importanza, aumentando costantemente il numero di dispositivi connessi che vengono utilizzati per automatizzare processi, raccogliere dati e fornire servizi a livello domestico, commerciale e industriale. Tuttavia, a causa della vasta gamma di tecnologie, protocolli e standard utilizzati, la creazione di un ecosistema integrato è risultato ostile.
Per risolvere questo problema si è deciso di sviluppare un nuovo protocollo unico chiamato Matter; esso punta a creare un'infrastruttura unificata per la connettività dei dispositivi IoT. Gli obiettivi di Matter sono quelli di garantire l'interoperabilità fra i vari dispositivi, rendere la configurazione e la gestione più semplice e garantire un elevato livello di sicurezza nella trasmissione dei dati.
In questa tesi si darà un'idea delle potenzialità offerte da Matter per la creazione di un ecosistema integrato di dispositivi IoT e dei vantaggi che ne possono derivare in termini di semplicità di configurazione, sicurezza e interoperabilità.

Indice

Introduzione

L'Internet of Things (IoT) sta diventando sempre più importante nel mondo di oggi, con un numero sempre maggiore di dispositivi connessi che vengono utilizzati per automatizzare processi, raccogliere dati e fornire servizi a livello domestico, industriale e commerciale. Essi possono essere rappresentati mediante una rete di dispositivi connessi che interagiscono fra di loro attraverso uno scambio di informazioni. A costituire questo settore vi sono diverse tipologie di apparecchiature come elettrodomestici, sensori, veicoli, prodotti per la casa e molto altro.

Lo sviluppo dell'IoT nell'ultimo periodo è sempre più rapido, tutto ciò è dovuto all'aumento della disponibilità di connessioni ad internet e dall'utilizzo di avanzate tecnologie di comunicazione wireless. Questo ha permesso ai dispositivi di diventare sempre più intelligenti e interconnessi, migliorando la qualità della vita delle persone e rendendo possibile l'implementazione di nuove soluzioni e servizi. La continua espansione di questo settore sta generando nuove opportunità di business e sfide per le aziende che cercano di sfruttare tutto il potenziale della tecnologia.

Per poter sfruttare tale potenziale però è necessario trovare delle soluzioni alle difficoltà che si presentano. La mancanza di standard condivisi e unificati rende difficile la creazione di soluzioni interconnesse e interoperative. Essi generano delle difficoltà nella scelta del protocollo adeguato alla specifica soluzione, poiché deve garantire un'ottima gestione e compatibilità tra i dispositivi, oltre alla sicurezza dei dati trasmessi. Per risolvere questi problemi si è deciso di sviluppare un protocollo unico chiamato Matter. Esso viene promosso da un consorzio di aziende leader nel settore, tra cui Google, Amazon, Samsung, Apple e altre.

I principali obiettivi che si pone Matter sono quelli di fornire un'infrastruttura unificata per l'IoT. L'idea è quella di creare un protocollo che sia in grado di supportare una vasta gamma di dispositivi, fornendo un'interfaccia comune per la gestione e la comunicazione. Inoltre, Matter punta a risolvere i problemi di interoperabilità e compatibilità tra i diversi sistemi IoT, semplificando l'adozione e l'utilizzo di dispo-

sitivi IoT per gli utenti finali. Infine il protocollo si concentra sulla sicurezza e sulla privacy, fornendo un livello di protezione adeguato per i dati sensibili gestiti.

La tesi è così organizzata:

- Capitolo 1: Viene analizzata l'architettura di Matter, soffermandosi brevemente sulle specifiche dei protocolli sottostanti, e si spiega il funzionamento a livello di rete. Viene poi illustrato il modello di dati sulla quale si basa e il modello di interazione attraverso la quale i dispositivi comunicano. Infine vengono mostrati i principali ambiti di applicazione disponibili e futuri.
- Capitolo 2: Si analizza la sicurezza del protocollo, spiegando i principi sulla quale si basa sia a livello di sicurezza che di privacy, di seguito si parla della categorizzazione delle minacce e delle relative contromisure, infine si parla dell'architettura da un punto di vista di sicurezza, andando a vedere le certifiche dei dispositivi.
- Capitolo 3: Questo capitolo mostra l'implementazione di Matter, ossia la sua messa in servizio, spiega il concetto di ecosistemi e della funzionalità Multi-Admin ed infine mostra le possibili integrazioni con ulteriori protocolli IoT.

Capitolo 1

Fondamenti e Funzionamento di Matter

1.1 Origini e sviluppo del protocollo

L'idea di sviluppo di un protocollo unificato ebbe inizio nel 2019 con la nascita di una partnership chiamata "Project Connected Home Over Ip" (Project CHIP). Tale protocollo punta allo sviluppo di un nuovo standard che permette la semplificazione sotto molteplici aspetti quali produttività degli apparati smart, scelta dei dispositivi idonei, configurazione degli apparati e della rete, ecc...
L'idea nasce fra le più grandi aziende tecnologiche al mondo (Google, Amazon, Apple, Tuya e molte altre) le quali, mediante un'azienda inizialmente nota come Zigbee Alliance e poi trasformatasi in CSA (Connectivity Standards Alliance), decisero la creazione di un nuovo standard per poter risolvere i problemi creati negli ultimi anni. Questo standard si pone l'obiettivo di garantire l'intercambiabilità dei device e l'interoperabilità all'interno della rete. Tale progetto con il passare degli anni è stato formalizzato ed ha assunto il nome di Matter.
Le caratteristiche fondamentali di questo protocollo sono:

- **Semplicità:** facilità nella scelta dei dispositivi, nella configurazione e nella gestione.
- **Interoperabilità:** possibilità di scegliere diversi produttori per l'implementazione di dispositivi smart all'interno della propria abitazione.
- **Affidabilità:** utilizzo di tecnologie innovative e stabili, che permettano l'utilizzo dei dispositivi in qualsiasi condizione.

- **Sicurezza:** attraverso l'implementazione di controlli di sicurezza efficaci, crittografia avanzata dei messaggi e connessioni crittografate end-to-end.
- **Flessibilità:** possibilità di gestire i device della rete da diversi dispositivi contemporaneamente, andando a formare quindi diversi ecosistemi interoperativi.

1.2 Architettura

Matter punta a costruire un protocollo di comunicazione universale basato su IPv6 per i dispositivi intelligenti. Il protocollo è composto da diversi livelli applicativi che verranno implementati sui dispositivi, oltre a ciò i diversi livelli di collegamento saranno indispensabili per mantenere l'interoperabilità.

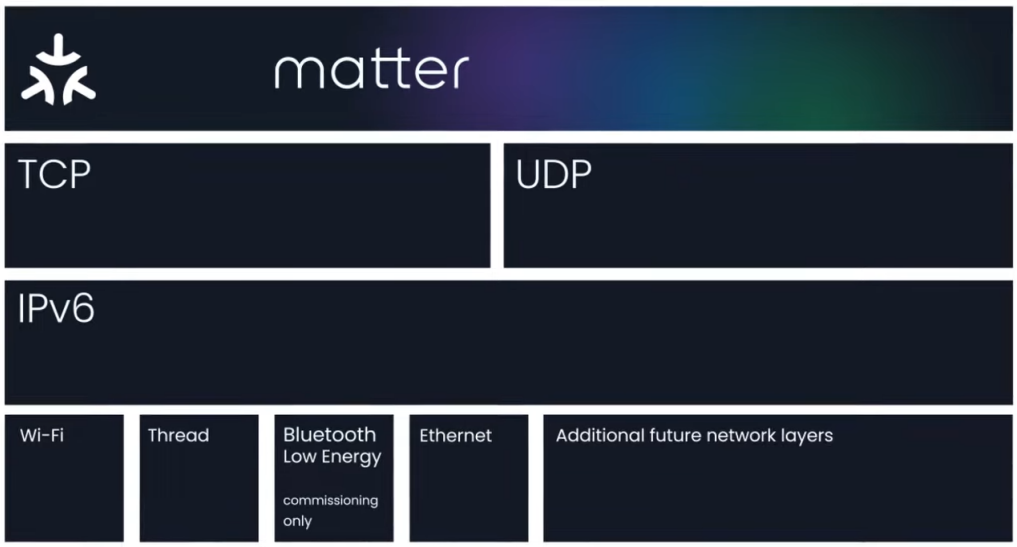

Figura 1.1: Architettura Matter [1]

Notiamo subito come Matter opera ad alto livello, ossia si avvale di tutte le tecnologie sottostanti per poter garantire all'utente la massima semplicità ed interoperabilità, pur mantenendo alta la sicurezza.
Matter si basa quindi su protocolli già esistenti, fra questi i più importanti sono: Wi-fi, Ethernet, Thread e Bluetooth Low Energy (BLE).
Ognuno di questi verrà spiegato al fine di poter comprendere meglio la struttura del protocollo ed avere quindi una visione più chiara delle possibilità che si hanno nell'implementare una rete Matter.

1.2.1 Wi-fi / Ethernet

I protocolli Wi-fi ed Ethernet possono essere raggruppati in quanto svolgono la stessa mansione. Essi vengono impiegati all'interno dell'ecosistema per andare a gestire quei dispositivi che necessitano di una banda di trasmissione elevata o quei dispositivi i quali non possono richiedere l'utilizzo di un hub per connettersi alla rete come ad esempio gli elettrodomestici.
Inoltre con l'avvento di Wi-fi 6 diventa molto interessante l'utilizzo di tale protocollo in quanto risulta progettato per l'ambito IoT andando ad aumentare il numero di dispositivi connessi, riducendo la latenza nella trasmissione dei dati ed ottimizzando il consumo di energia per i dispositivi alimentati a batteria.

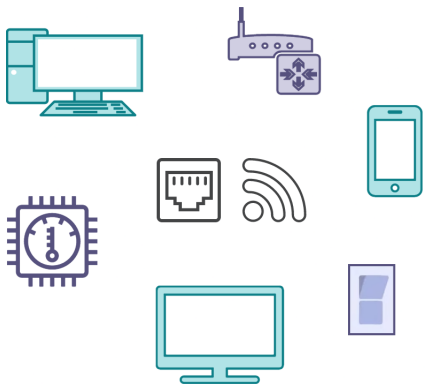

Figura 1.2: I dispositivi sono connessi tutti alla stessa radice [1]

Le connessioni possibili attraverso l'uso del Wi-Fi possono avvenire mediante l'estensione su più segmenti, a condizione che siano collegati tutti ad un dispositivo comune che svolga la funzione di Access point.

1.2.2 Bluetooth Low Energy (BLE)

L'utilizzo della tecnologia BLE avviene sono in determinati casi, essa viene infatti adoperata principalmente durante la fase di Commissioning, ossia durante l'inizializzazione del device. Permette di associare rapidamente un dispositivo ad una rete, andando così ad evitare le attuali procedure lunghe e complesse. Tale argomento verrà approfondito nei prossimi capitoli.

1.2.3 Thread

Lo scopo di Thread è gestire la connessione dei dispositivi mediante l'uso di una tecnologia a bassa potenza che non richiede bande di trasmissioni ampie.
Thread è quindi un protocollo di reti mesh wireless a basso consumo energetico nato

nel 2014. Viene usato principalmente per i dispositivi che necessitano di efficienza energetica. Si basa su IPv6 e usa lo standard di comunicazione IEEE 802.15.4 senza fili a corto raggio. Essendo una rete mesh i dispositivi sono in grado di comunicare fra loro ed adattarsi ad eventuali variazioni di rete, risolvendo eventuali guasti.

Vi possono essere principalmente due tipologie di dispositivi: il router (o anche detto Full Thread Device, FTD) e i dispositivi finali (Minimal Thread Device, MTD). Queste tipologie vi sono affinché si mantenga una solidità della rete, con i router che avranno i ruoli di controllori e gestori del traffico.

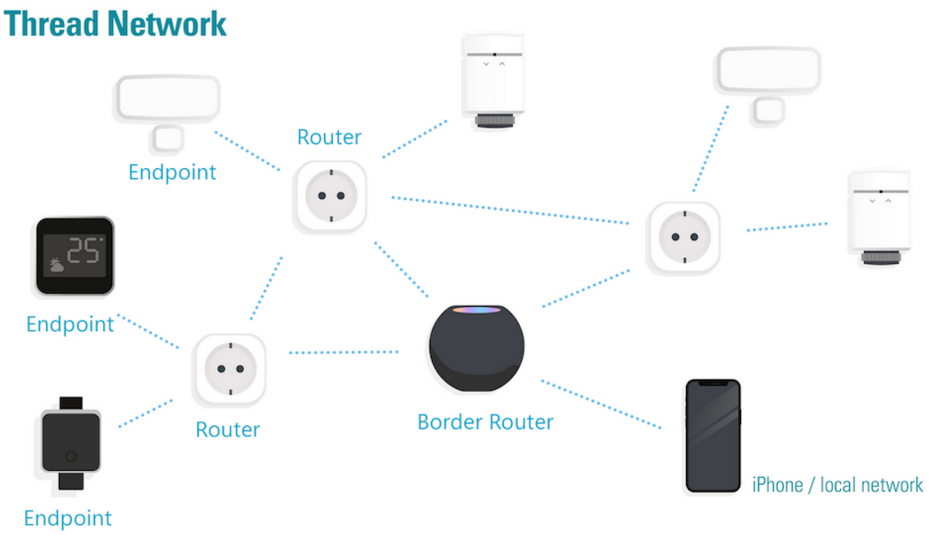

Figura 1.3: Esempio di rete Thread [14]

Dalla figura possiamo intuire che i dispositivi in grado di svolgere il ruolo di Router sono quelli che hanno sempre una connessione energetica attiva, ad esempio luci smart, prese, etc...

Gli FTD (Router) possono operare nella rete sia come Router sia come Dispositivo Finale, in qualità di router hanno la capacità di effettuare servizi di routing tra i dispositivi Thread, permettono l'aggiunta di nuovi dispositivi e gestiscono la sicurezza della rete.
Gli MTD (Dispositivi finali o anche detti Endpoint) dialogano solo con i Router e non possono inviarsi messaggi fra di loro. Un MTD si collega ad un solo Router per volta e trasmette dati in modo temporizzato, riducendo così il consumo energetico.

Durante la creazione della rete, i Dispositivi Finali scelgono un Router e se un routing risulta ridondante, allora il Router in questione si declassa autonomamente a Dispositivo Finale. Fra tutti i Router vi è sempre uno che funge da leader, ma qualora dovesse subire dei malfunzionamenti verrà subito sostituito da un altro FTD.

La rete Thread non comunica in modo diretto con le reti adiacenti. Per poter comunicare è necessaria la presenza di un Border Router, ossia un Router di Confine che faccia da "interprete". Esso è fondamentale per poter interagire con una rete Thread. Oltre a svolgere il ruolo di Router fornisce connettività IP end-to-end dalla rete Thread alle reti adiacenti, come Wi-fi o Ethernet.

Notiamo come l'interazione mediante dispositivi quali smartphone risulta impossibile senza la presenza di un Router di Confine, poiché l'interazione sarebbe possibile solo se anche il telefono avesse tale protocollo integrato. Per poter interagire mediante smartphone è quindi necessaria la presenza di un ponte di passaggio.

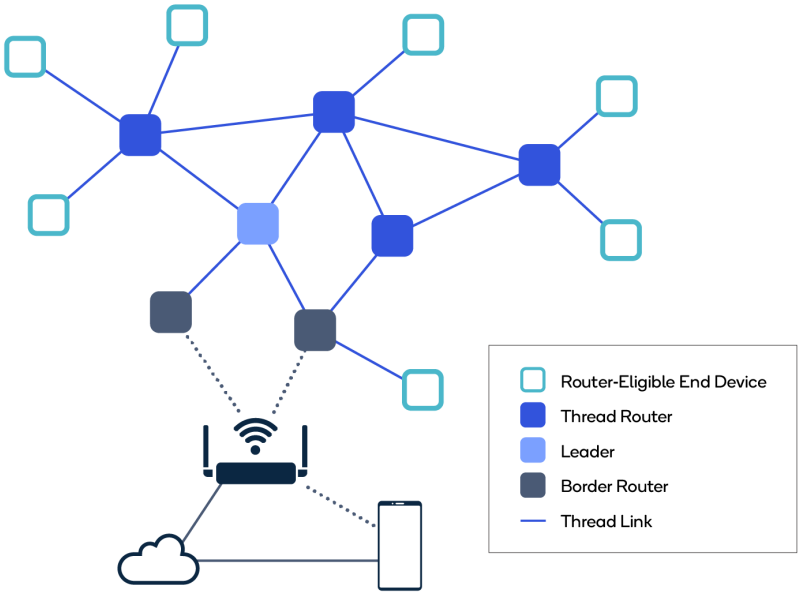

Figura 1.4: Topologia di rete Thread completa [13]

Il problema principale di questa tecnologia dovuta all'assenza di standard è l'eventuale presenza di più Border Router; in questo caso si possono avere degli scenari dove non tutti i dispositivi possono connettersi allo stesso Border Router e non tutti i Border Router vengono visualizzati dallo stesso device, non visualizzando quindi

una porzione di rete. Tali problematiche però sono state risolte con il rilascio di un aggiornamento il quale permette che qualsiasi dispositivo Thread possa connettersi a qualsiasi Router di Confine, tralasciando il produttore. Questo anticipa ciò che viene poi standardizzato in Matter e quindi l'unificazione di tutte le connessioni fra dispositivi.[8]

1.3 Funzionamento del protocollo a livello di rete

Una delle caratteristiche fondamentali di questo protocollo riguarda la connettività alla rete esterna. Essendo basato su un intreccio di connessioni interne ed essendo sviluppato con la concezione di una elevata affidabilità e semplicità di utilizzo, esso non necessita di una connessione ad un cloud per poter essere sfruttato.

Solitamente l'interazione dei dispositivi domotici all'interno di una rete avviene sempre mediante uno scambio di informazioni con il cloud, ossia l'utente richiede un determinato stato di un dispositivo al cloud, il cloud lo processa e lo invia al dispositivo, il quale una volta letto imposta lo stato richiesto.

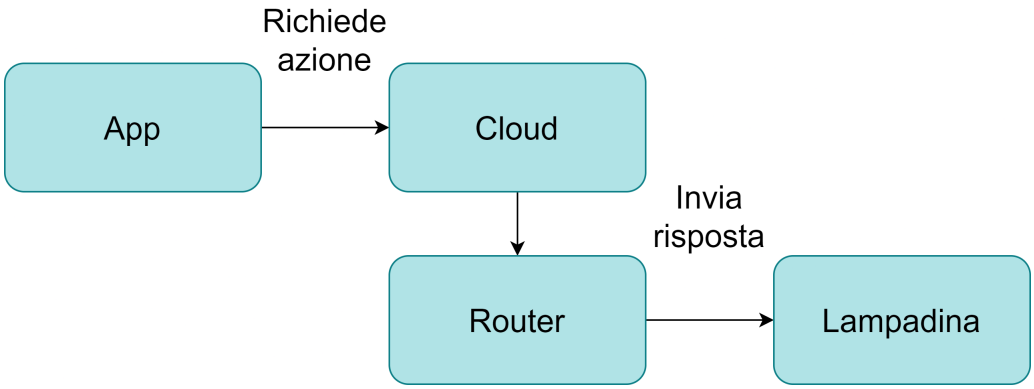

Figura 1.5: Esempio di connessione in cloud

Questo non permette una continua operatività dei dispositivi in quanto in assenza di accesso ad internet essi non sarebbero più utilizzabili.

Per risolvere questo problema Matter opera in locale, pertanto non vi è più la necessità di essere collegati ad un cloud ma tutte le sue funzioni possono essere svolte all'interno della propria rete. Questo rappresenta un enorme vantaggio in termini di velocità d'esecuzione e di risposta, sicurezza e privacy ed anche riguardo l'affidabilità dei dispositivi.

Una comunicazione tipo viene effettuata nel seguente modo: l'utente richiede un determinato stato ad un dispositivo, il dispositivo riceve subito la richiesta e l'utente può verificare immediatamente lo stato del dispositivo. Il passaggio avviene tutto internamente senza richiedere nessun accesso ad internet.

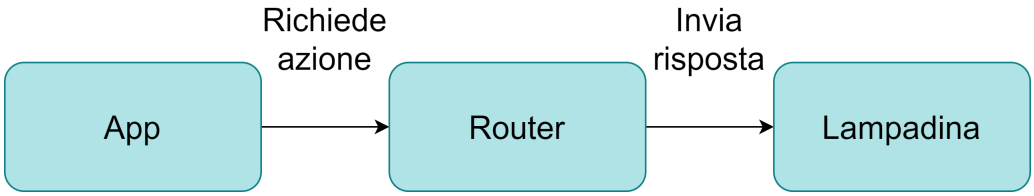

Figura 1.6: Esempio di connessione in locale

Dopo aver illustrato le differenze fra i protocolli usati in precedenza, andiamo ora a vedere una composizione tipo di una rete connessa interamente in Matter.

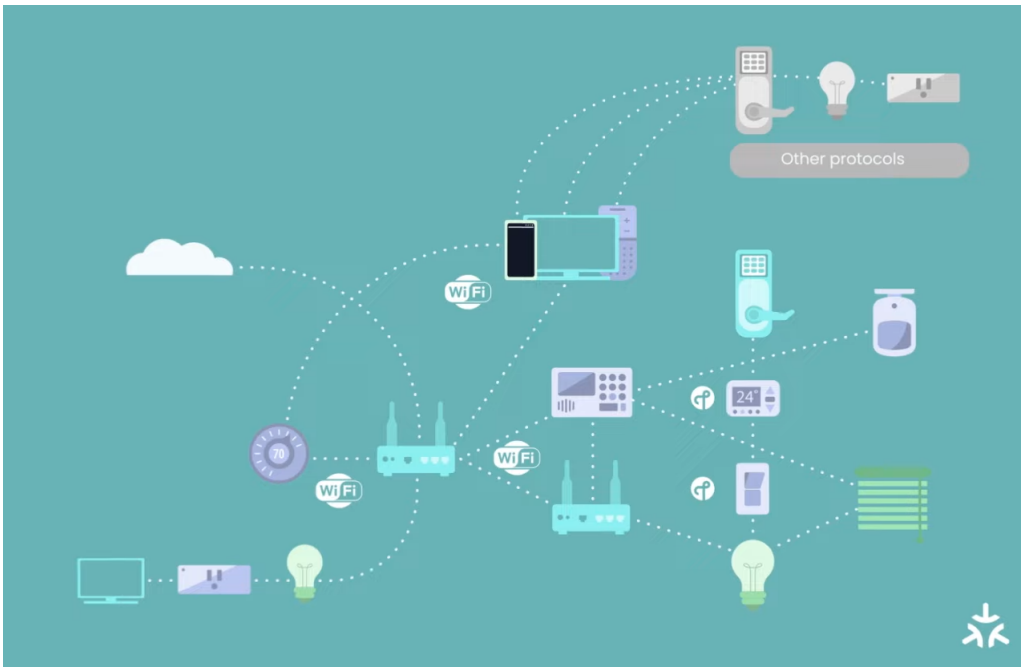

Figura 1.7: Rete completa Matter [1]

La rete illustrata mostra come possono avvenire le varie connessioni e mostrano anche un dettaglio particolare che approfondiremo nei prossimi capitoli, ossia la connessione ed interoperabilità con protocolli esterni a Matter.
Matter è stato pensato per risolvere tutte le problematiche dovute all'utilizzo di diversi protocolli, pertanto per potersi insediare si è pensato ad un metodo che consentisse un'integrazione efficace e semplice.

1.4 Data Model: gli elementi fondamentali

Il modello di dati di Matter definisce gli elementi che compongono un nodo tipico di Matter. Entrando nell'ambito di sviluppo, in genere si esprimono le capacità del dispositivo attraverso il seguente modello di dati.

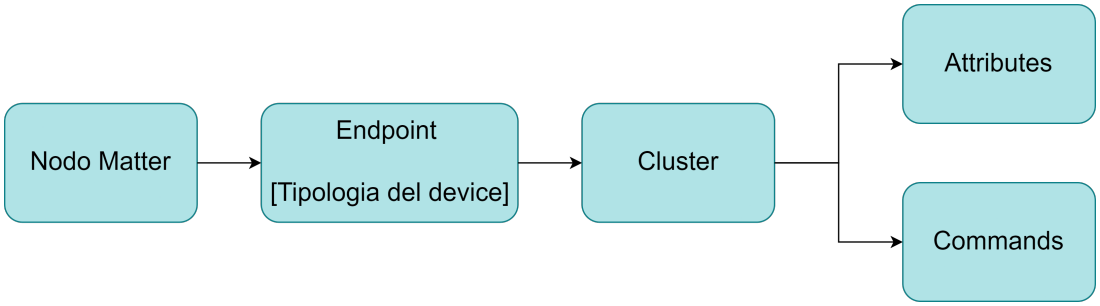

Figura 1.8: Rete completa Matter [15]

Tutti i dispositivi sono composti da Nodi[1]. Un nodo rappresenta un'entità mediante un indirizzo di rete unico che espone delle funzionalità. La comunicazione di rete ha origine e termina sempre a livello di un nodo. All'interno dei nodi vi possono essere più Endpoint, ognuno di essi include un set di funzionalità. Ad esempio supponendo un dispositivo di illuminazione, un nodo potrebbe rappresentare la gestione della luce mentre un endpoint potrebbe rappresentare una modalità di gestione, *On/Off*.

1.4.1 Cluster

All'interno di un endpoint vi possono essere uno o più Cluster. Essi raggruppano specifiche funzionalità come la gestione *On/Off* su una presa smart o il *controllo del livello* su una lampadina dimmerabile. Diversi endpoint possono avere istanze della stessa funzionalità, ad esempio una luce potrebbe controllare singolarmente se stessa oppure potrebbe gestire un'insieme di luci.
Un Cluster è composto da Attributi, Comandi ed Eventi.

Attributi

Gli Attributi rappresentano gli stati detenuti dal nodo, ad esempio il livello attuale di un cluster di controllo del livello. Essi possono essere persistenti o volatili e anche di sola lettura o lettura/scrittura. Possono infine essere definiti attraverso diversi tipi di dati, come integers, floats, boolean, strings o array.

[1]La specifica Matter determina che un dispositivo può avere più nodi. Ad esempio, gli smartphone possono avere più app, ognuna delle quali è un nodo diverso. Di conseguenza, tutti i dispositivi conterranno un singolo nodo. La maggior parte dei dispositivi fisici seguirà questo modello.[11]

Comandi

I Comandi sono azioni che possono invocare dei comportamenti specifici. Ogni comando può avere dei parametri associati ed equivalgono all'invio di messaggi diretti di procedure da eseguire come *"spegnere la luce"* o *"sbloccare la serratura"*.

Eventi

Infine gli Eventi sono dei record delle transizioni passate dello stato. Rappresentano quindi le azioni svolte in precedenza e consentono di acquisire informazioni utili a comprendere i passaggi di stato del dispositivo fino ad arrivare allo stato attuale.

Tipologie di dispositivi

Nel complesso notiamo come i cluster possano variare di dispositivo in dispositivo e quanto essi siano riutilizzabili. Tale funzionalità risulta molto comoda agli sviluppatori in quanto seppur lo sviluppo di un cluster possa richiedere tempo, esso verrà poi risparmiato poiché tale funzionalità sarà estendibile a qualsiasi device.
La scelta dei cluster viene definita nella libreria dei cluster di applicazioni, consultabile su un documento associato alla documentazione ufficiale di Matter.

Server e Client

Ogni Cluster detiene una parte server e una parte client. Sebbene un server sia stateful e contenga attributi, eventi e comandi, un client è stateless ed ha il compito di avviare interazione con un cluster server.

Le operazioni che può eseguire sono le seguenti: lettura e scrittura negli attributi del server, lettura degli eventi presenti nel server e chiamata dei comandi remoti. Per poter definire meglio il concetto di Server e Client possiamo pensare ad un rapporto fra chi richiede un servizio e chi lo offre. Quando un dispositivo offre un servizio verrà inteso come server, quando invece richiede un servizio allora svolgerà la funzione di client.

Ogni Nodo può avere al suo interno diversi endpoint contenenti tipologie di cluster diversi, un esempio può essere il seguente:
Avendo due dispositivi di illuminazione (lampade da tavolo) ognuna di esse detiene due endpoint, l'illuminazione della lampadina e l'interruttore che gestisce l'accensione. Notiamo quindi che il comportamento degli endpoint possiede un ruolo diverso a seconda dell'azione che svolge/riceve.

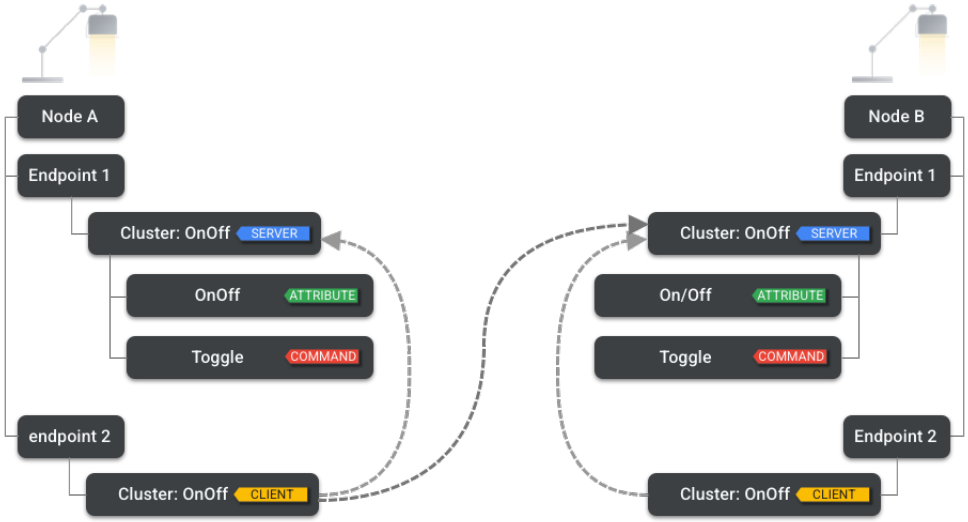

Figura 1.9: Funzionamento completo di un nodo comprensivo di endpoint, cluster, attributi e comandi [11]

Quando si parla di accensione della luce, allora ci si sta rivolgendo alla luce direttamente, la quale non avendo possibili interazioni svolge la funzione di server, essa infatti può subire azioni. Quando si parla invece di azionamento dell'interruttore ci si sta rivolgendo all'interruttore, il quale può svolgere l'interazione di azionare la luce, pertanto rappresenta la funzione di client.
Nella figura osserviamo la presenza di due lampade da tavolo le quali, se azionati i loro interruttori, forniscono azioni diverse. L'interruttore A agirà sulla lampada A e B mentre l'interruttore B agirà sulla lampada B.

1.4.2 Endpoint 0

Mentre fin'ora abbiamo visto come i cluster di applicazioni possano essere implementati su qualsiasi endpoint andiamo ora a porre attenzione sull'endpoint 0. L'endpoint 0 non è utilizzabile per le funzioni normali in quanto è riservato ai cluster di utilità. I cluster di utilità sono cluster specifici che racchiudono le funzionalità di servizio su un endpoint, ad esempio rilevamento, indirizzamento, diagnostica ed aggiornamento software.

1.5 Interaction Model: come comunicano i nodi

Mentre il modello di dati definisce la composizione di un nodo, il modello di interazione definisce i suoi comportamenti. Senza il modello di interazione un nodo non può eseguire operazioni. I nodi possono interagire fra di loro nei seguenti modi:

- Lettura e iscrizione ad Attributi ed Eventi
- Scrittura in Attributi
- Richiamo dei comandi

Per poter generare un'interazione é necessario che i nodi stabiliscano delle connessioni criptate fra loro. All'interno di una interazione vi possono essere più Transazioni di diverse tipologie. Ogni Transazione è a sua volta composta da più Azioni che possono essere intraprese come messaggi a livello di messaggistica immediata tra i nodi. Queste interazioni possono quindi essere ridefinite e scomposte in Transazioni di lettura, scrittura e di chiamata.

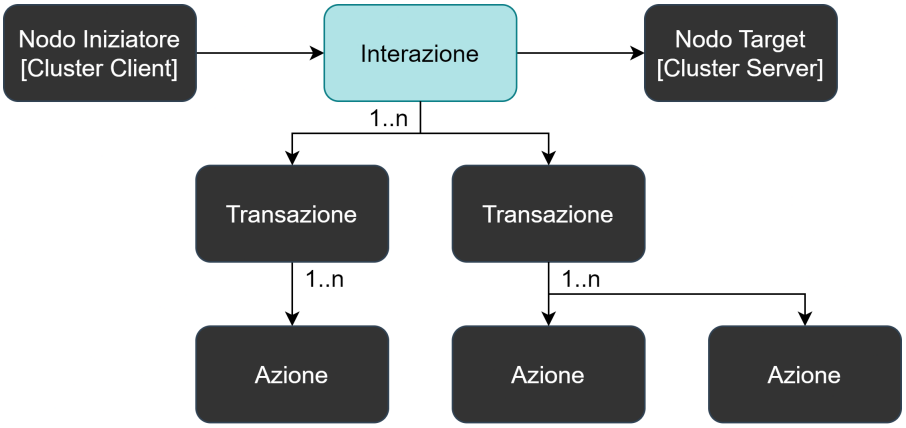

Figura 1.10: Composizione interazione dal nodo iniziale al nodo finale.

Nella figura notiamo come il nodo iniziale prende il nome di *Iniziatore*, quello finale invece *Target*. Ad ogni nodo corrisponde un Cluster che può essere definito a sua volta Client o Server in base alla mansione svolta. Solitamente l'iniziatore è sempre un Cluster Client mentre il target è sempre un Cluster Server ma in una transazione particolare (Transazione con abbonamento) i ruoli potrebbero invertirsi.

Più nodi possono comunicare fra loro contemporaneamente mediante i Gruppi. Un Gruppo di dispositivi è un meccanismo implementato per inviare messaggi simultaneamente a diversi dispositivi. Tutti i nodi appartenente al gruppo possiedono un *GroupID*. Questa implementazione segna un passo avanti in quanto in precedenza

l'invio contemporaneo a più device di un'azione generava sempre un effetto popcorn, ossia in base all'ordine d'invio dei messaggi verso i dispositivi essi venivano eseguiti in momenti diversi, generando latenza e ritardi nella trasmissione dei dati. L'utilizzo di questa nuova funzionalità invece permette l'esecuzione in contemporanea su tutti i dispositivi. Per poter effettuare questa comunicazione è necessario oltre al GroupID l'utilizzo della tecnologia Multicast fornita dalla tecnologia IPv6.

Per la comunicazione fra nodi è necessario specificare sempre il percorso che deve seguire l'interazione. Va tenuto conto inoltre della casistica in cui ci si trova, quindi se si sta comunicando con un nodo singolo o con un gruppo.
A seconda della scelta di comunicazione si possono avere le seguenti implementazioni:

```
<path> = <node> <endpoint> <cluster> <attribute | event | command>
<path> = <group ID>        <cluster> <attribute | event | command>
```

Figura 1.11: Implementazione tipo del percorso. [10]

Alcuni tipi di transazioni possono tenere conto di un eventuale Timeout o meno. Questa implementazione è stata necessaria per la prevenzione di possibili attacchi *MITM (Man In The Middle)*.
Risulta particolarmente efficace per i dispositivi che accedono all'accesso di risorse come serrature. Di seguito si spiega brevemente come può avvenire un attacco *MITM* e come il ruolo del tempo possa incidere su tale minaccia.
Un attacco di intercettazione ha il seguente pattern:

1. Alice invia a Roberto un messaggio iniziale, ad esempio un'azione di richiesta di scrittura.
2. Eva, un uomo in mezzo, intercetta il messaggio e impedisce a Roberto di riceverlo, ad esempio attraverso alcuni tipi di Jam radio.
3. Alice, non ricevendo una risposta da Roberto, invia un secondo messaggio.
4. Eva intercetta di nuovo e impedisce a Bob di riceverla.
5. Eva invia il primo messaggio intercettato a Roberto, come se provenisse da Alice.
6. Bob invia la risposta ad Alice (ed Eva).

7. Eva tiene il secondo messaggio intercettato per una replica successiva. Poiché Bob non ha mai ricevuto il secondo messaggio intercettato originale da Alice, lo accetterà. Questo messaggio rappresenta una violazione della sicurezza quando codifica un comando, ad esempio "Apri blocco".

Per evitare questo tipo di attacchi, le Azioni a tempo impostano un timeout massimo per la transazione al loro inizio. Anche se Eva riuscisse a eseguire i primi sei passaggi del vettore di attacco, non potrà riprodurre il messaggio nel passaggio 7 a causa di un timeout scaduto sulla transazione.
Le transazioni a tempo aumentano la complessità e il numero di azioni. Di conseguenza, non sono consigliate per ogni transazione, ma solo le operazioni fondamentali sui dispositivi che hanno il controllo sugli asset per la privacy e la sicurezza fisici o virtuali.[10]

1.5.1 Transazione di lettura

La transazione in lettura consiste nella richiesta di dati da un dispositivo. Questa tipologia di transazione può essere scomposta in 3 azioni:

- **Read Request Action**
- **Report Data Action**
- **Status Response Action**

In seguito andremo ad approfondire ognuna di queste azioni.

Read Request Action

Verso di direzione: Iniziatore → Target
L'iniziatore esegue una query su uno specifico target inserendo gli attributi e/o eventi richiesti. Dopo aver ricevuto la richiesta di lettura dal target, assembla un report contenente tutti i dati richiesti.

Report Data Action

Verso di direzione: Target → Iniziatore
Il target una volta ricevuta la richiesta risponde con le seguenti informazioni:

- Report sugli attributi
- Report sugli eventi

- Richiesta di interruzione risposta: un flag che determina se la risposta di stato deve essere presente o meno.
- ID abbonamento: se il report fa parte di una transazione di abbonamento deve includere l'identificativo.

Status Response Action

Verso di direzione: Target → Iniziatore → Target
L'azione di risposta dello stato avviene solo quando l'iniziatore ha ricevuto i dati richiesti. Attraverso questa azione si conferma la ricezione dei dati segnalati. Qualora il flag di interruzione della risposta è impostato, l'iniziatore non è tenuto ad inviare una risposta. Al termine di tale azione la query può considerarsi completata, essa serve principalmente per garantite la riuscita dell'operazione.

Transazione in abbonamento

In questa casistica le operazioni subiscono delle modifiche, essenzialmente l'iniziatore non richiederà una singola risposta ma risposte periodiche. Oltre a fornire i dati per la generazione della query, l'iniziatore (detto abbonato) dovrà fornire anche l'intervallo minimo e massimo secondo la quale vuole ricevere risposte. Il target (detto editore) invia il primo report e resta in attesa di una risposta dello stato. Una volta ricevuta inizierà l'invio periodico da parte sua di report verso l'abbonato fino a quando l'azione non sarà annullata o andrà persa.
Vi sono delle limitazioni tuttavia nell'esecuzione di tale transazione:

- Le azioni richieste agiscono solo in Unicast
- Tutte le azioni devono avere lo stesso ID abbonamento
- Se l'abbonato non riceve report entro i tempi stabiliti, allora l'abbonamento viene terminato

1.5.2 Transazione di scrittura

La transazione in scrittura consiste nella modifica di un valore di un attributo in un cluster. Questa tipologia di transazione può avvenire senza un timeout o con, prendono quindi il nome di transazione di scrittura non-programmata e transizione di scrittura a tempo.

Non-programmata

Questa tipologia non prevede un tempo entro il quale ricevere una risposta, ed è composta da due azioni: Azione richiesta di scrittura e Azione risposta di scrittura.

Nella richiesta abbiamo il seguente verso di direzione: Iniziatore → Target
Come per la transazione di lettura, l'iniziatore fornisce al target le richieste di scrittura, un'eventuale flag per la richiesta a tempo e uno per l'interruzione della risposta.

Nella risposta invece abbiamo la seguente direzione: Target → Iniziatore
Dopo che la destinazione ha ricevuto la richiesta di scrittura, completa la transazione con una risposta che include un report contenente le scritture effettuate e i relativi errori se presenti.

A tempo

Le transazioni a tempo possiedono un passaggio in più rispetto alle transazioni non programmate. La prima richiesta che effettuata viene chiamata Azione di richiesta tempo.

Essa avviene nel seguente verso: Iniziatore → Target
L'iniziatore invia una richiesta dove specifica il numero di millisecondi in cui la transazione può restare aperta. Al termine di quel periodo l'azione scade e non potrà più essere considerata valida. Una volta ricevuta tale azione il target invierà una risposta per poter convalidare l'informazione ricevuta.
Successivamente viene svolta la transazione nello stesso modo di quella non-programmata.

1.5.3 Transazione di chiamata

Le transazioni di chiamata vengono utilizzate per richiamare uno o più comandi di cluster su un nodo di destinazione. Anch'esse supportano le transazioni a tempo e non a tempo. Possono essere suddivise in due azioni: Richiamo azione di richiesta e Richiamo azione di risposta.

Richiamo azione di richiesta

Verso di direzione: Iniziatore → Target
Come per le richieste di lettura e scrittura, anche in questo caso l'iniziatore fornirà dei dati al target, tali dati sono un elenco di percorsi nei comandi del cluster, eventuali flag per le richieste a tempo e per l'interruzione delle risposte, ed un ID

interazione, un numero usato per abbinare l'azione di richiesta di chiamata all'azione di risposta di chiamata.

Richiamo azione di risposta

Verso di direzione: Target → Iniziatore
Una volta che la destinazione ha ricevuto la richiesta, esso finalizzerà la transazione con una risposta che porta un elenco di risposte di comandi o stato per ogni richiesta inviata e l'ID interazione associato.

Richiamo di richiesta e risposta a tempo

Anche in questo caso le richieste potrebbero pervenire a tempo, tuttavia il meccanismo resta lo stesso di quanto spiegato per le transazioni di scrittura a tempo.

1.6 Ambiti di applicazione del protocollo Matter

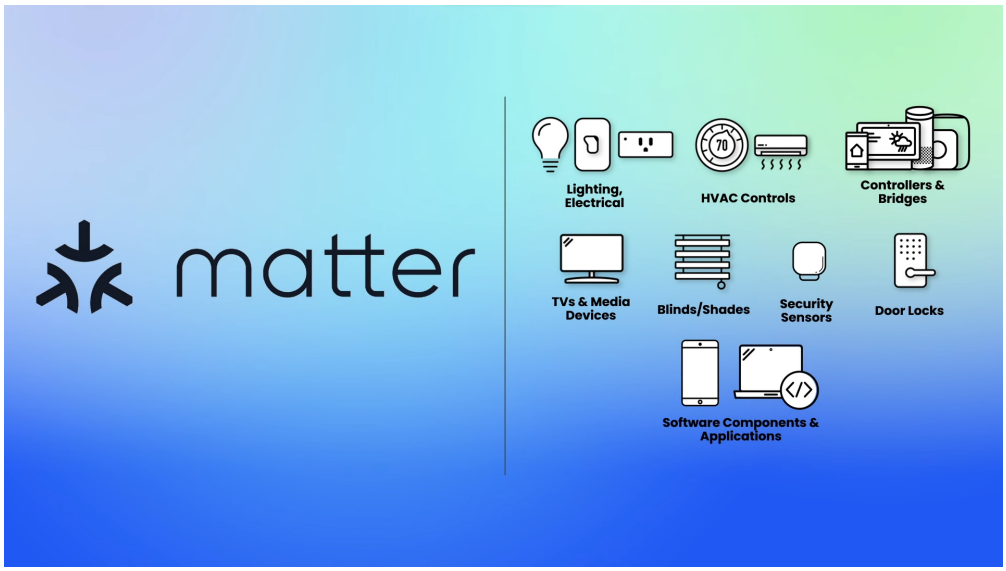

Figura 1.12: Aree attualmente sviluppate e già compatibili con dispositivi Matter [5]

Attualmente Matter non è ancora disponibile in tutti i device ma risulta implementato solo per le categorie mostrate nella figura.

Tali categorie comprendono la maggior parte dei dispositivi più usati e diffusi al momento. Si sta tuttavia continuando ad implementare ulteriori aree per poter fornire un supporto sempre maggiore. Il rilascio di nuovi device supportati, di nuove funzionalità e miglioramenti risulta essere programmato ogni 2 anni.

Questo per consentire uno sviluppo di alta qualità e per fare in modo di rilasciare un prodotto funzionante e certificato, compatibile con qualsiasi ecosistema.

Di seguito si mostrano le aree attualmente in sviluppo che verranno rilasciate in futuro.

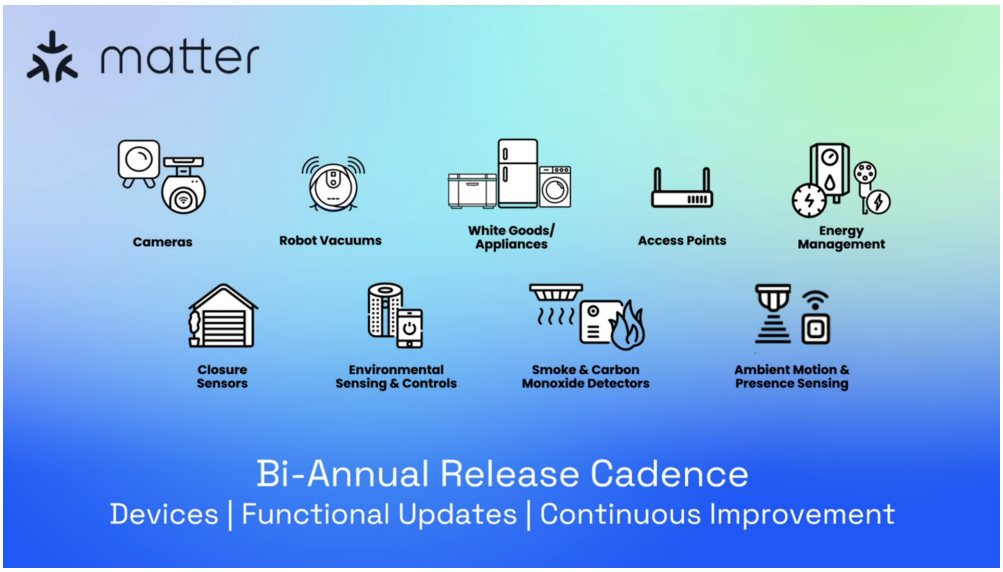

Figura 1.13: Prossime aree attualmente in sviluppo [5]

Capitolo 2

Sicurezza integrata

Con l'introduzione sempre maggiore di dispositivi all'interno delle abitazioni, crescono sempre di più i problemi legati ai cyber attacchi. Sfruttando vulnerabilità di vario genere essi riescono a causare danni all'utente finale. In questo capitolo verrà affrontato nel dettaglio l'argomento sicurezza e privacy, andando ad elencare tutte le caratteristiche del protocollo Matter al fine di illustrare quanto esso sia efficace sotto questo aspetto.

Gli attacchi cyber hanno come scopo principale l'infezione dei dispositivi al fine di generare diverse tipologie di rischi come: [3]

- Malfunzionamento del prodotto attraverso il controllo remoto
- Attacchi DDoS (Distributed Denial of Service)
- Violazione dei dati e della privacy
- Furto della proprietà intellettuale
- Potenziale danno alle persone

Per poter contenere queste minacce sono state sviluppate delle best practice per i dispositivi mobili, i pc e il cloud che possono essere impiegate anche in contesti IoT e sono:

- Verifica dell'identità del device / Autenticità del dispositivo comprovata
- Comunicazione sicura
- Controllo degli accessi

Una sfida importante per i produttori è l'implementazione di queste best practice su larga scala, come ad esempio il provisioning sicuro di un'identità in un dispositivo della casa intelligente, il controllo sicuro di un dispositivo da remoto, l'aggiornamento del software e il reset del device, ecc.
Gli aspetti chiave da affrontare durante l'implementazione della sicurezza nelle case intelligenti includono:

1. **Produzione:** I dispositivi possono essere soggetti ad una serie di attacchi come l'iniezione di codice maligno durante la fase di produzione.

2. **Operazione:** Una volta installati i dispositivi sono soggetti ad una vasta gamma di attacchi, sia da remoto che a livello fisico.

3. **Manutenzione:** Per poter garantire una sicurezza costante ed efficace è necessario il continuo aggiornamento dei dispositivi, tuttavia l'esecuzione dell'aggiornamento deve provenire da canali sicuri per evitare caricamenti di software non autorizzati che potrebbero compromettere il device.

Per mitigare queste vulnerabilità è sufficiente seguire le seguenti pratiche usate in ambito cybersecurity:

- **Produzione sicura:** Questo comprende il provisioning dell'identità e del firmware del dispositivo in modo sicuro presso strutture fidate.

- **Comunicazione e operazioni sicure:** Le comunicazioni devono essere crittografate e autenticate per proteggersi da attacchi remoti e locali. La protezione deve avvenire garantendo l'arrivo dei dati in forma riservata, autenticata e inalterata.

- **Aggiornamenti via etere:** Deve essere garantito il supporto riguardante gli aggiornamenti di sicurezza dei dispositivi, del firmware e delle credenziali per poter aumentare la protezione verso nuove minacce.

- **Regolamenti di sicurezza:** La regolamentazione dell'IoT per la sicurezza e la privacy variano di paese in paese, pertanto è necessario determinare delle linee comuni al fine di garantire una protezione elevata per chiunque.

Per poter contenere quanto detto fino ad adesso Matter si basa su 5 principi di sicurezza e 6 principi di privacy, di seguito andremo a spiegare brevemente ognuno di esso.

2.1 Principi di sicurezza

2.1.1 Completo

Matter risulta completo in quanto si basa su 3 caratteristiche:

- Approcci a livello basati su autenticazione e attestazione per la messa in servizio.
- Protezione di ogni messaggio mediante una crittazione end-to-end.
- Aggiornamenti firmware garantiti seguendo la tecnologia over-the-air.

La sicurezza funzionale di Matter dunque risulta autonoma, ossia non dipende dalla sicurezza delle tecnologie sottostanti attraverso la quale viene eseguito. Inoltre essendo definito in un framework open-source, facilita l'aderenza alle specifiche ed aumenta l'interoperabilità tra i diversi sistemi.

2.1.2 Forte

Matter utilizza una serie di metodi di sicurezza all'avanguardia. Agisce mediante una serie di tecniche crittografiche ben collaudate e riconosciute.
Utilizza AES nelle seguenti modalità:

- CCM per la riservatezza e l'integrità con chiavi a 128 bit.
- CTR è utilizzato per proteggere gli identificatori e preservare la privacy.

SHA-256 è utilizzato per l'integrità ed ECC con la curva "secp256r1" per le firme digitali e gli scambi di chiavi, schemi di derivazione delle chiavi standard e generatori di numeri casuali.

Oltre a questa suite crittografica, Matter si affida a protocolli standard di sessione basati su passcode e di creazione di certificati per stabilire sessioni sicure per l'inserimento, l'attestazione e il funzionamento.
Inoltre, Matter adotta il concetto di attestazione del dispositivo, ossia un dispositivo Matter non può entrare a far parte di un Fabric[1] se non si dimostra autentico.

[1]Per Fabric si intende un insieme di dispositivi che condividono la stessa trusted root (radice di fiducia). La radice di fiducia in questione è la Root CA che emette i NOCs (Node Operational Credentials Specification: rappresenta un insieme di credenziali che permettono ad un nodo di identificarsi con il Fabric.) alla base delle identità dei nodi. All'interno di ogni Fabric ogni nodo è identificato in modo univoco da un Node ID stabile. La selezione e l'assegnazione di questi costrutti all'interno di Matter garantisce l'unicità degli identificatori e fornisce una guida chiara sulla proprietà e la gestione degli spazi dei nomi.

Per garantire un ecosistema uniforme e conforme, Matter estende anche la tecnologia CSA Distributed Compliance Ledger ai dispositivi Matter per fornire una piattaforma interoperabile che consente ai dispositivi Matter commissionati di verificare se gli altri dispositivi Matter sono stati certificati.

2.1.3 Facile da usare

La sicurezza di Matter è progettata per rendere i dispositivi intelligenti più facili da implementare per i produttori di dispositivi e da utilizzare per i consumatori.
Per ogni aspetto funzionale della sicurezza, la specifica di base di Matter include esempi e vettori di test. Le implementazioni di riferimento di Matter, disponibili per tutti i produttori attraverso un repository GitHub, comprendono un'implementazione software della sicurezza funzionale di Matter composta da moduli separati. I clienti che acquistano i dispositivi Matter non dovranno preoccuparsi della sicurezza perché è già presente.

2.1.4 Resistente

La sicurezza di Matter è resistente: è progettata per proteggere, rilevare e recuperare. Matter offre più di un modo per eseguire determinate operazioni. Per mostrare quanto affermato é possibile fare un esempio riguardo un tentativo di attacco DDoS, ogni messaggio scambiato fra i nodi possiede un contatore ed esso può essere usato come contro misura all'attacco.
Oltre a questo esempio un altro può essere l'introduzione del DCL, Distributed Compliance Ledger. Esso contiene diverse informazioni riguardanti il produttore e il venditore ed essendo decentralizzato, si avvale della tecnologia blockchain, non possiede alcun server centrale, garantendo così una continua autenticità. Attraverso esso è possibile consultare la versione più recente dei firmware installati ed eventuali aggiornamenti da effettuare.

2.1.5 Agile

Infine risulta Agile, non c'è sicurezza senza aggiornamenti, ed aggiornare i sistemi non sempre è possibile. Per questo Matter grazie alla sua flessibilità crittografica può affrontare nuovi sviluppi e minacce. Esso è costruito in modo da poter implementare nuove misure di sicurezza nelle sue versioni future senza dover subire drastiche modifiche. Il design modulare dei protocolli consente una rapida sostituzione al fine di garantire una sicurezza elevata.

2.2 Principi di privacy

La privacy dei dati è un requisito importante per tutti i sistemi che gestiscono informazioni personali e Matter non fa eccezione. La privacy dei dati è radicata in Matter e funge da concetto fondamentale per tutti i protocolli e i metodi di interazione. [4]

Aderisce al Regolamento generale sulla protezione dei dati (GDPR) e alle successive normative sulla privacy in altre parti del mondo. Oltre a questo pone dei principi riguardo la materia di privacy dei dati, tali principi sono i seguenti:

- **Confidenzialità ed Integrità:** Matter utilizza il più alto livello possibile di standard crittografici civili per le comunicazioni di rete, per garantire che entità non autorizzate non possano facilmente accedere o manomettere i dati comunicati tra i dispositivi Matter.
- **Prova d'identità:** Richiesta per i dispositivi Matter con certificati crittografici, in modo che i dati siano condivisi solo tra entità Matter conosciute.
- **Standard aperto:** Consente a chiunque di ispezionare il modello per le interazioni di Matter tra nodi legittimi.
- **Minimizzazione dei dati:** I dati condivisi nell'ambito delle interazioni di Matter sono ridotti al minimo, riducendo così il potenziale di fuga involontaria di informazioni.
- **Scopo definito:** I dati condivisi tra i nodi Matter sono strettamente finalizzati a uno scopo definito, ovvero alle operazioni specifiche dei dispositivi come richiesto dal protocollo Matter.
- **Meccanismi di tutela della privacy:** Uso della crittografia per garantire che i messaggi o le identità delle parti comunicanti non avvengano in chiaro sulla rete.

La completa aderenza a questi principi richiede non solo il supporto dei protocolli standard Matter, ma anche di tutto l'ambiente e l'infrastruttura di supporto in cui i dispositivi Matter vivono e operano. Questo perché l'obiettivo è proteggere la privacy degli individui. Matter dal suo canto non gestisce direttamente informazioni rilevanti per l'uomo ma solo informazioni riguardanti i dispositivi. Per quanto riguarda la gestione dei dati indirettamente collegati a informazioni personali attraverso correlazione o inferenza, applica tutti i principi espressi in precedenza.

2.3 Threat Model

Con l'aumento continuo di dispositivi per l'IoT, aumentano anche i rischi di attacchi e compromissioni informatiche. Per far fronte a questa minaccia, come abbiamo visto in precedenza, la sicurezza e la privacy sono i principi chiave della progettazione della materia. Utilizzando tali principi di progettazione, di seguito analizzeremo le minacce e le misure di contenimento che potremmo adottare per affrontare queste tipologie di attacco. Per poter valutare le minacce è necessario individuare gli attori, il perimetro di attacco e le contromisure applicabili.

Per individuare gli attori bisogna tenere a mente vari fattori come:

- **Motivazioni:** Indica il fine dell'attacco, può essere personale, politico, finanziario.
- **Capacità:** Ogni attaccante viene classificato in base alle capacità che possiede, tali capacità comprendono risorse, abilità e accesso.
- **Ruolo:** L'attaccante potrebbe essere un ospite, in questo caso dispone di accesso fisico ai dispositivi e potrebbe manomettere la configurazione attuale al fine di ottenere dati.
- **Ciclo di vita:** Il rischio di possedere un dispositivo manomesso potrebbe verificarsi quando è di seconda mano, non si ha la garanzia che il prodotto sia al suo stato originale.

Prendendo come riferimento il perimetro di attacco, bisogna tenere conto degli obiettivi e dei metodi da sfruttare.

Gli obiettivi a cui si fa riferimento contengono dispositivi abilitati a Matter che gestiscono informazioni sensibili come sensori, allarmi o qualsiasi dispositivo che contenga delle vulnerabilità.
I metodi fanno riferimento alle vulnerabilità che un dispositivo può presentare e alle azione che ne possono conseguire.
Un esempio di dispositivo vulnerabile potrebbe essere il gateway e il metodo per generare l'attacco potrebbe essere lo sniffing[2] del traffico.

Di seguito si riportano tutti i possibili target:

- **Dispositivi:** Sensori, allarmi, controller, etc...

[2]Si definisce l'attività di intercettazione passiva dei dati che transitano in una rete.

- **Rete:** Mesh, locale, wireless, cablata, etc...
- **Gateway:** Router, Bridge, etc...
- **Servizi:** Cloud, servizi di aggiornamento, servizi di sicurezza, etc...
- **Dati:** Archiviati, in transito, etc...
- **Fattore umano:** Utenti, ospiti, persone fidate, etc...
- **Protocolli:** Matter, Thread, Wi-fi, etc...
- **Algoritmi:** Difetti di progettazione, vulnerabilità, etc...

e i relativi metodi di attacco

- **Dispositivi:** Sfruttare vulnerabilità, etc...
- **Rete:** Sniffing, modifica e disturbo della rete, etc...
- **Gateway:** Infezioni, attaccare il traffico o i nodi dietro di esso
- **Servizi:** Compromettere la CA o altri servizi critici
- **Fisico:** Attacco fisico al dispositivo o ad altri componenti
- **Fattore umano:** Ingannare o influenzare esseri umani fidati ad attaccare
- **Protocolli:** Trovare vulnerabilità nei protocolli
- **Algoritmi:** Individuare e sfruttare le vulnerabilità dell'algoritmo

Le minacce possono infine essere analizzate tenendo conto di diversi aspetti come:

- **Severità:** la gravità della minaccia ci permette di decidere l'importanza da attribuirgli ed è composta da una combinazione di probabilità e impatto.
- **Probabilità**, basata su
  - **Accesso:** Indica come esso può avvenire, da remoto oppure fisico o in prossimità.
  - **Difficoltà:** Valuta la creazione della minaccia, ossia quanto è in grado di diffondersi e quanto impatti sulla vittima.
- **Impatto**, basato su

- **Scopo:** Indica in raggio di azione dell'attacco, se ad un singolo dispositivo, ad una intera rete o ad una parte di un ecosistema.
- **Controllo dei dati:** Indica l'accesso dei dati ottenuti, può variare da dati di bassa sensibilità fino a dati strettamente confidenziali che potrebbero indicare una completa compromissione del dispositivo/rete.

Ad ogni minaccia corrisponde una relativa contromisura. Attualmente si contano più di 200 minacce rilevate. Spesso una singola contromisura risulta efficace verso diverse minacce ed è possibile consultare l'elenco completo nelle specifiche ufficiali di Matter. Di seguito verrà esposto un esempio al fine di mostrare l'effettiva sicurezza.

Descrizione della minaccia						
ID	Descrizione	Threat agent	Impatto/Consequenza	Severità	Impatto	Probabilità di successo
T59	Un messaggio maligno sfrutta la vulnerabilità del dispositivo, causandone la compromissione	Attaccante usando un dispositivo connesso alla rete	Il dispositivo di fiducia potrebbe essere compromesso	Alta	Alta	Alta

Contromisura nelle specifiche Matter	
CM58	I dispositivi supportano gli aggiornamenti del firmware OTA. I dispositivi convalidano l'autenticità e l'integrità del firmware prima dell'installazione.

Figura 2.1: Esempio di minaccia e relativa contromisura [2]

In questa casistica notiamo come la threat faccia riferimento ad un messaggio malevolo che sfrutta una vulnerabilità. Essa può avvenire mediante un dispositivo connesso alla rete, e la conseguenza risulta una possibile infezione. Viene considerata una minaccia a score alto in quanto un dispositivo compromesso potrebbe significare una possibile esfiltrazione di dati oltre che ad una compromissione di più device all'interno della rete.
La contromisura associata risulta essere il supporto dell'aggiornamento del firmware OTA. Essendo un dispositivo soggetto ad aggiornamenti sarà più difficile comprometterlo in quanto il sistema risulta sempre protetto dalle ultime vulnerabilità scoperte.

2.4 Architettura di sicurezza

Per poter garantire un'elevata sicurezza Matter include diversi livelli di protezione. Primo fra tutti è l'utilizzo del concetto di identità in modo da poter garantire al consumatore la sicurezza del dispositivo introdotto all'interno dell'abitazione. Ogni dispositivo possiede una certificazione che attesta la sua identità e che protegge dai falsi. La certificazione comprende diverse informazioni facenti riferimento al produttore, al venditore e all'oggetto in se. Per poter comprendere come la certificazione venga generata è bene fare un passo indietro e definire le sue basi. Matter basa il suo modello di sicurezza sulle PKI, ossia le infrastrutture a chiave pubblica. Di seguito verrà approfondito il concetto al fine di comprendere come tale tecnologia risulti sicura.

2.4.1 Public Key Infrastructure

La Public Key Infrastructure (PKI) è un sistema di sicurezza crittografico basato su chiavi pubbliche e private utilizzato per garantire l'autenticità, l'integrità e la riservatezza dei dati scambiati su una rete. La PKI utilizza un insieme di componenti software, protocolli e standard per creare, gestire, distribuire e revocare le chiavi crittografiche pubbliche e private necessarie per autenticare e proteggere i dati.

In pratica, la PKI funziona creando un sistema a due chiavi: una chiave pubblica e una chiave privata. La chiave pubblica viene distribuita al pubblico, mentre la chiave privata è mantenuta segreta dal proprietario. Un mittente crittografa i dati utilizzando la chiave pubblica del destinatario, e solo il destinatario può decifrare i dati utilizzando la propria chiave privata.

La PKI è utilizzata per una vasta gamma di applicazioni, tra cui la crittografia dei dati in transito su una rete, la protezione dei dati archiviati e la gestione dell'identità digitale. Ad esempio, la PKI viene spesso utilizzata per autenticare il sito web di una banca o di un negozio online, garantendo che i dati dell'utente siano protetti durante la trasmissione. La PKI viene anche utilizzata per la firma digitale dei documenti, garantendo che il documento sia autentico e non sia stato modificato dall'originale.

La PKI richiede l'utilizzo di una catena di certificati, in cui i certificati sono rilasciati da autorità di certificazione (CA) di fiducia. Una CA è un'organizzazione che verifica l'identità di un'entità e rilascia un certificato digitale che attesta l'iden-

tità verificata. I certificati possono essere revocati in caso di furto di chiavi o di compromissione del certificato stesso.

2.4.2 Come vengono generati i certificati

Ora che abbiamo introdotto le PKI passiamo all'analisi dei certificati di Matter. Ogni dispositivo Matter possiede un DAC, Device Attestation Certificate.
Tale certificato è definito mediante il rilascio di diverse attestazioni.
Il DCL, Digital Compliance Ledger, di cui abbiamo già parlato in precedenza è alla base di tutto. Esso è un database basato su blockchain, quindi su rete decentralizzata ed è composto da 5 database, di seguito si riportano i nomi comprensivi delle loro funzioni:[7]

- **Vendor Schema:** Fornisce informazioni generali su un fornitore, come il nome legale dell'azienda, il nome del marchio preferito associato al VendorID, l'URL della pagina di riferimento del fornitore, ecc.
- **Device Model Schema:** Fornisce informazioni generali su un dispositivo, come ProductName, Product ID, PartNumber, Commissioning info, ecc.
  Queste informazioni sono condivise tra tutte le versioni software del prodotto.
- **Device Software Version Model Scheme:** Fornisce informazioni specifiche sulla versione del software, ad esempio URL delle note di rilascio, FirmwareDigest, URL dell'immagine software OTA, ecc.[3]
- **Compliance Test Result Schema:** Fornisce dati sulla conformità e sui risultati dei test del dispositivo.
- **PAA Schema:** Fornisce un elenco dei certificati delle autorità di attestazione dei prodotti per le PAA approvate.

Attraverso il DCL notiamo il PAA, Product Attestation Authority. Esso rappresenta una delle autorità di certificazione di sicurezza coinvolte nel protocollo Matter. Il PAA ha il compito di verificare che i produttori di dispositivi IoT abbiano seguito le specifiche di sicurezza richieste dal protocollo per i loro prodotti. Si assicura quindi che i dispositivi siano stati progettati e prodotti in modo sicuro e che soddisfino gli standard di sicurezza di Matter.

[3]Viene memorizzato solo l'URL in DCL, quindi i fornitori dovranno memorizzare le immagini OTA nella propria sede e pubblicare solo l'URL dell'immagine in DCL.

I produttori per ottenere un certificato di attestazione del prodotto, devono sottoporre i propri dispositivi a test rigorosi condotti da laboratori accreditati dal PAA. Questi test valutano la sicurezza nelle aree di crittografia, autenticazione, autorizzazione e integrità dei dati. Una volta superati i test ed ottenuta tale certificazione, subentra una nuova entità, il PAI o anche detto Public Attestation Intermediate.

Il ruolo del PAI è quello di verificare che i certificati di attestazione del prodotto risultino validi e che i dispositivi soddisfino gli standard di sicurezza di Matter, emettendo ulteriori certificati di sicurezza per quei dispositivi.

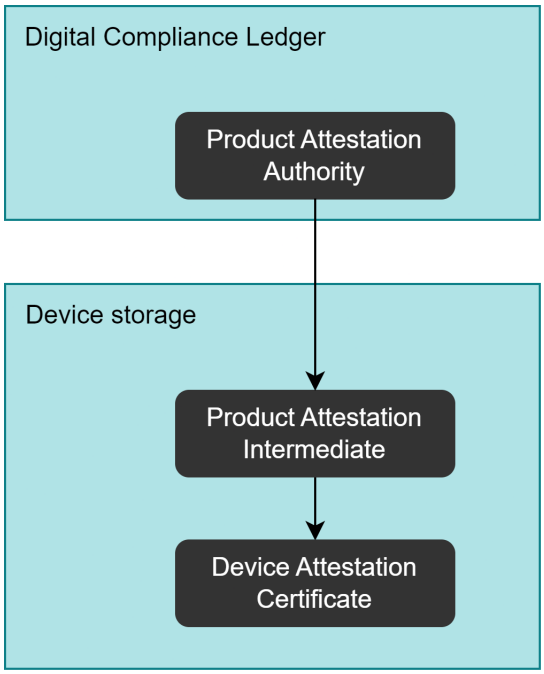

Figura 2.2: Nella figura è possibile vedere come esse siano rilasciate e a quale gruppo appartengano, ossia la PAA viene registrata a livello DCL mentre PAI e DAC vengono registrate a livello di dispositivo. [12]

Ottenuta la certificazione dal PAA e dal PAI si giunge così all'ultima certificazione ritenuta la più importante, il DAC.
Il Device Attestation Certificate (DAC) è un certificato di sicurezza che viene emesso per ogni dispositivo che aderisce al protocollo Matter. Esso rappresenta la prova che il dispositivo è stato verificato e autenticato come sicuro dal PAA. Il DAC è emesso dal PAI, che fa parte della catena di certificazione di sicurezza di Matter.
Esso è progettato per fornire un meccanismo di autenticazione forte per i dispositivi IoT. Contiene informazioni sul dispositivo, tra cui il nome del soggetto, il numero di serie, il modello e la versione del firmware. Inoltre, include la chiave pubblica del dispositivo, che viene utilizzata per verificare la sua firma digitale.

L'utilizzo del DAC nel protocollo Matter garantisce che i dispositivi siano autentici e sicuri. Quando uno di esso richiede la connessione ad una rete Matter, il DAC viene presentato come prova della sua identità e della sua sicurezza.

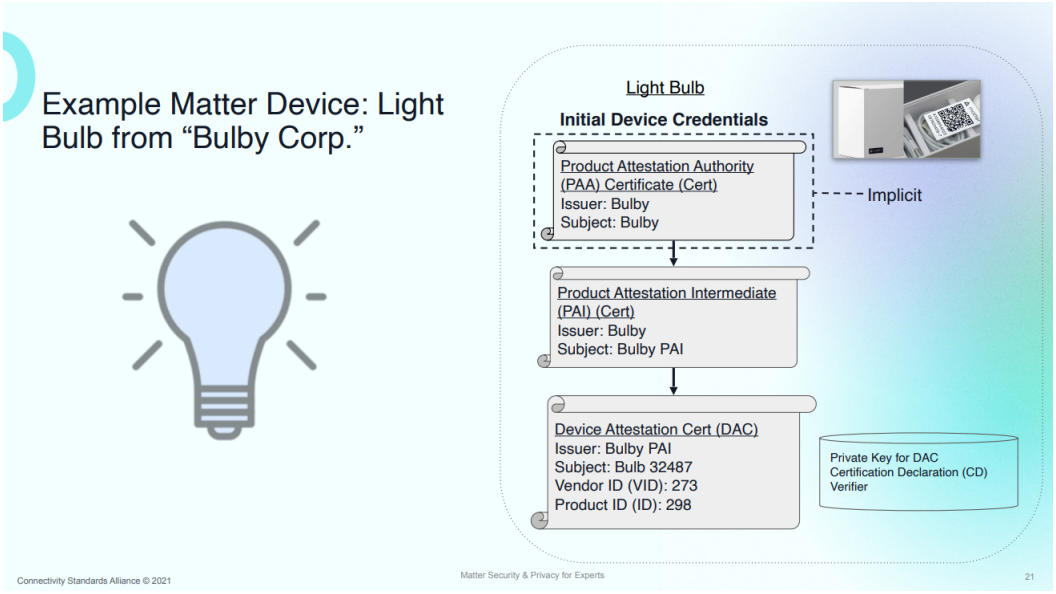

Figura 2.3: Esempio di una lampadina smart prodotta dall'azienda "Bulby Corp."[2]

Dopo aver visto l'utilizzo delle PKI per l'attestazione del device andiamo a vedere come essa viene impiegata nella certifica dei nodi attraverso lo sviluppo di una catena di certificazione di sicurezza che include la Root Certificate Authority (RCA), l'Intermediate Certificate Authority (ICA) e il Node Operational Center (NOC).

Il Node Operational Center (NOC) è un componente chiave dell'architettura di Matter che gestisce le chiavi di sicurezza dei dispositivi IoT e li registra nella rete in modo che i dispositivi possano autenticarsi e comunicare in modo sicuro. Esso fornisce anche servizi di gestione degli errori e di ripristino per i dispositivi IoT, garantendo la continua funzione anche in caso di problemi di rete. Viene utilizzato inoltre per gestire i certificati di sicurezza dei dispositivi Matter e garantire che siano aggiornati e validi. Infine il NOC fornisce un'interfaccia per la gestione e il monitoraggio delle reti Matter, consentendo agli utenti di configurare, monitorare e gestire le reti di dispositivi IoT Matter.
La sua attestazione avviene tramite la CA(Root Certificate Authority).

La Root Certificate Authority (CA) è un'autorità di certificazione di sicurezza di alto livello. È responsabile della creazione e della gestione di tali certificazioni, garantendo la sicurezza dei dispositivi e dei loro dati.

La Root CA emette certificati di sicurezza per i PAA, i PAI e i NOC, e ogni certificato emesso viene utilizzato per garantire l'integrità, la riservatezza e l'autenticità dei dati scambiati tra i dispositivi e la rete.

Il processo di emissione dei certificati di sicurezza da parte della Root CA è altamente sicuro e rigoroso, per garantire che i certificati vengano emessi solo ai dispositivi che soddisfano gli standard di sicurezza richiesti. Ciò include il controllo della conformità del dispositivo alle specifiche Matter, la verifica dell'identità del produttore del dispositivo e la conferma che il dispositivo abbia superato i test di sicurezza richiesti.

Inoltre emette certificati digitali per le ICA. Le ICA, Intermediate Certificate Authority sono entità di livello inferiore rispetto alla RCA. Ricevono da questa le chiavi di crittografia pubbliche e private, e utilizzano queste informazioni per emettere i certificati digitali utilizzati dagli utenti finali. Questi certificati vengono utilizzati per garantire la sicurezza dei dispositivi IoT nell'ambiente Matter e consentono loro di comunicare in modo sicuro con altri dispositivi e sistemi di rete.

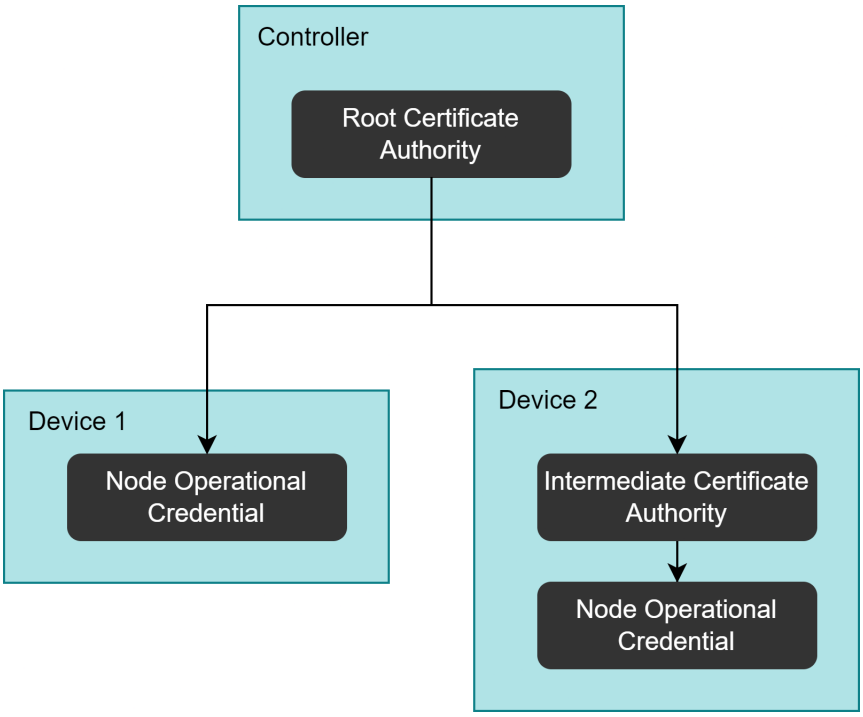

Figura 2.4: Modalità di rilascio del NOC[12]

Riepilogando, la connessione tra due dispositivi IoT nella rete Matter avviene attraverso l'utilizzo di certificati di sicurezza emessi dalla RCA e dall'ICA. Quando un dispositivo IoT si connette alla rete Matter, il certificato di sicurezza del dispositivo viene verificato dal NOC per garantire che il dispositivo sia autenticato e autoriz-

zato a connettersi alla rete. Inoltre, la crittografia end-to-end viene utilizzata per proteggere le informazioni trasmesse tra i dispositivi. Ciò garantisce che i dati siano protetti e che la comunicazione tra i dispositivi sia sicura e affidabile.

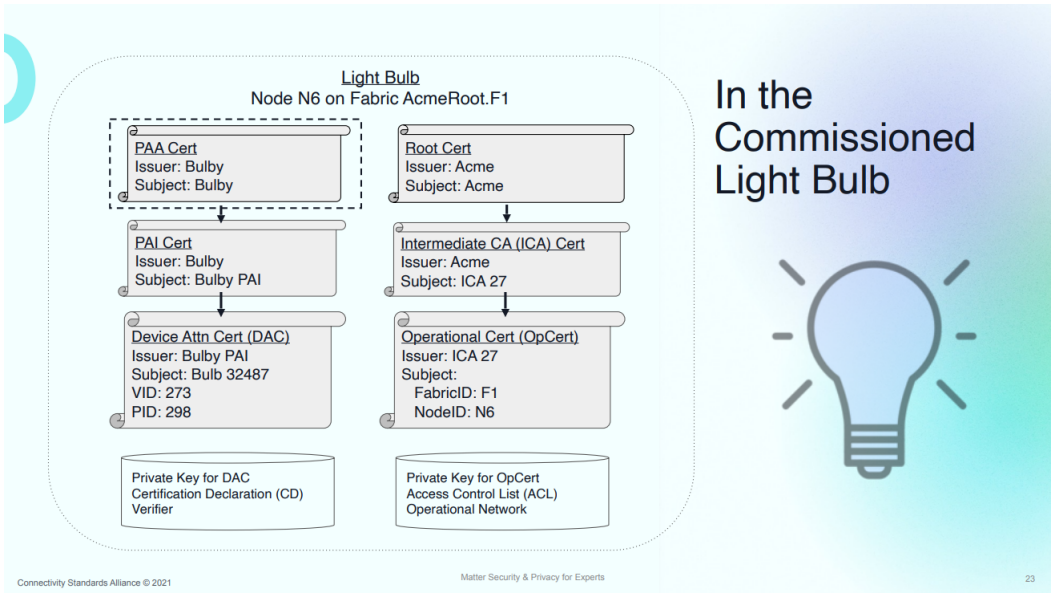

Figura 2.5: Dispositivo completo[2]

In figura è rappresentato un dispositivo finale, in questo caso la nostra lampadina di prima, completa di tutte le certificazioni necessarie per il funzionamento.

Concludiamo riassumendo quanto detto sulla sicurezza finora:

1. Messa in funzione del dispositivo facile, sicura e flessibile
2. Convalida dell'autenticità e della certificazione di ogni dispositivo
3. Informazioni aggiornate tramite il Registro di conformità distribuito (DCL)
4. Identità forte del dispositivo, in modo che solo i vostri dispositivi possano entrare a far parte della vostra casa intelligente
5. Comunicazioni Unicast protette
6. Comunicazioni di gruppo protette
7. Amministratori e controllori multipli, per massimizzare le possibilità di scelta
8. Controlli di accesso verificati per prevenire azioni non autorizzate
9. Aggiornamenti software standard e sicuri
10. Verifica dell'integrità del software

Capitolo 3

Implementazione e gestione di una rete Matter

Abbiamo visto in precedenza come è formato Matter e quali sono i suoi standard di sicurezza. Andremo a vedere ora come effettivamente il prodotto arrivi all'utente finale e come avviene la sua configurazione all'interno di una rete.
Verrà illustrato successivamente un'ulteriore funzione di sicurezza e gestione denominata Multi-Admin. Infine si parlerà dell'integrazione di Matter all'interno di diversi ecosistemi IoT già presenti, prendendo come esempio la tecnologia Zig-Bee.

Tenendo conto di quanto detto finora, sappiamo che un dispositivo giunge in negozio solo al termine di numerosi controlli eseguiti in precedenza. Ciò che contraddistingue Matter dagli altri dispositivi all'interno del negozio è la sua semplicità nella scelta. L'utente non dovrà più effettuare controlli sulle specifiche di compatibilità ma gli basterà controllare che sulla scatola del dispositivo sia presente il seguente simbolo.

Figura 3.1: Logo Matter presente su tutti i dispositivi compatibili.

Attraverso questo simbolo l'utente ha la garanzia che tale dispositivo rispetti i criteri imposti da Matter e non avrà bisogno di seguire guide o chiedere pareri tecnici per l'installazione, gli basterà seguire una procedura unica per tutte le tipologie dei dispositivi. Questo semplifica e incentiva l'utilizzo dei dispositivi smart da parte dei consumatori *"meno esperti"* del settore.
Di seguito si riporta una figura che spiega come inizializzare un dispositivo.

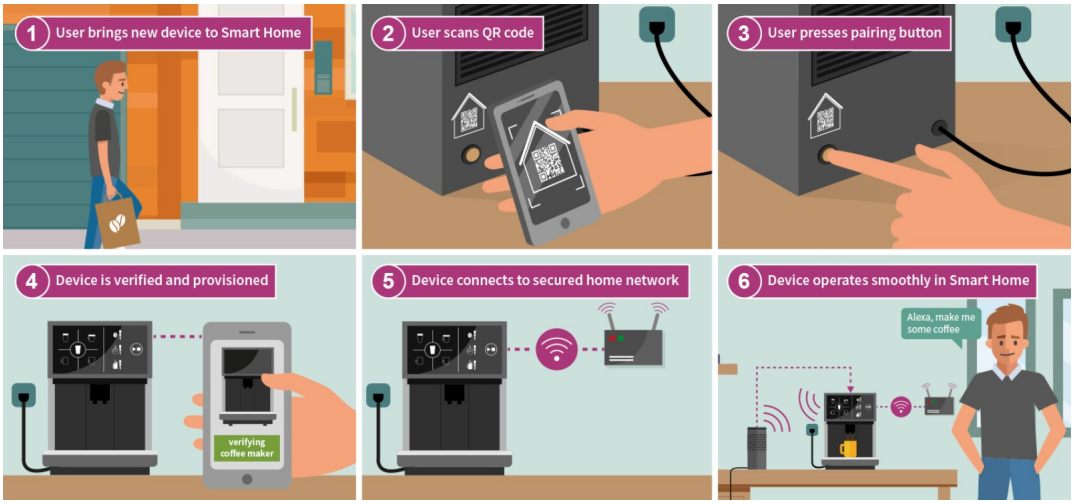

Figura 3.2: Messa in servizio di un dispositivo Matter da un utente[17]

Notiamo che nelle fasi non è più presente la necessità di scaricare diverse applicazioni dedicate e inserire all'interno i propri dati, ma è sufficiente un'app unica che supporti Matter. Questo snellisce tutte le procedure per la configurazione e rende il dispositivo utilizzabile fin da subito.

3.1 Commissioning di un dispositivo

Di seguito verrà mostrato cosa avviene esattamente all'interno dei dispositivi durante l'accoppiamento, tale procedura è denominata Commissioning.

Il Commissioning è una fase critica nella configurazione di un dispositivo IoT. In questa fase, il dispositivo viene accoppiato con la rete e viene configurato per funzionare correttamente all'interno di essa.

Il processo di Commissioning prevede l'utilizzo di un dispositivo mobile come un telefono o un tablet dotato di un'applicazione Matter compatibile (commissario). L'applicazione viene utilizzata per impostare i parametri di configurazione del dispositivo e per connetterlo alla rete Matter.

Durante la fase di Commissioning, il dispositivo e l'applicazione mobile si scambiano informazioni attraverso il protocollo Matter. In particolare, l'applicazione invia comandi di configurazione al dispositivo, mentre il dispositivo invia notifiche di stato all'applicazione per segnalare il completamento delle attività di configurazione. Una volta completata la fase di Commissioning, il dispositivo sarà in grado di comunicare con gli altri dispositivi della rete e di interagire con la piattaforma di automazione domestica o dell'edificio a cui è collegato.

Di seguito sono illustrati e spiegati i singoli passaggi effettuati fra il dispositivo e il commissario: [9]

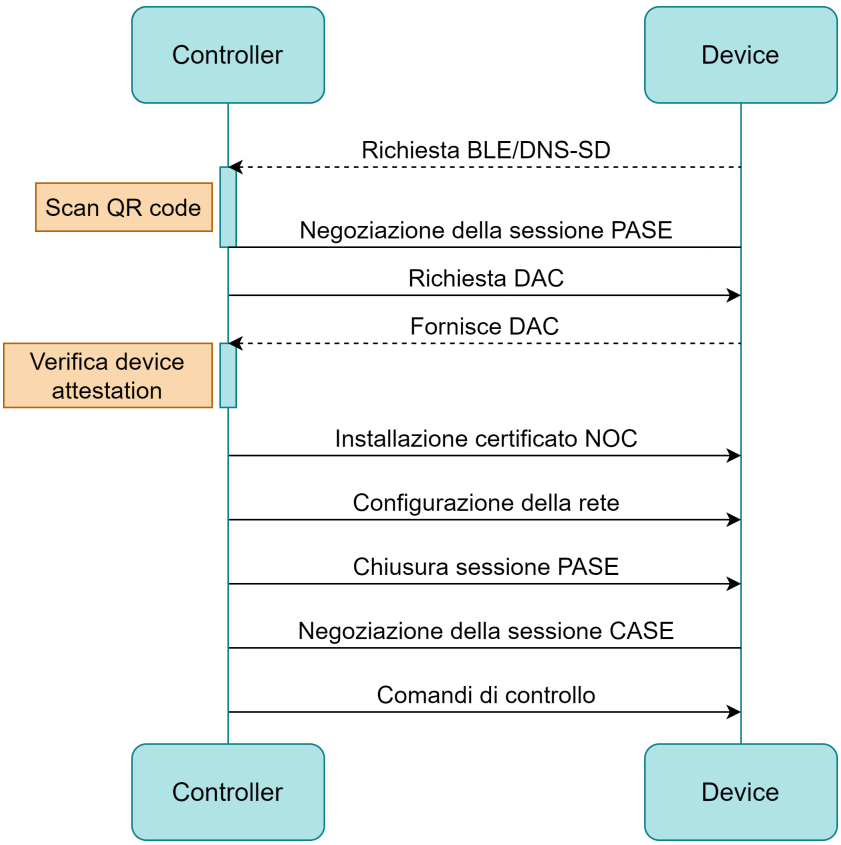

Figura 3.3: Flusso di commissione - Alto livello[12]

1. **Rilevamento dispositivi:** Il dispositivo utilizza i metodi a disposizione (Wi-fi, BLE, Thread) per poter essere rilevato dal Controller.
2. **Connessione al dispositivo (PASE):** Una volta che Controller e Device hanno trovato una corrispondenza, il commissario (controller) utilizza il passcode del payload di onboarding per eseguire la connessione **Passcode Authenticated Session Establishment (PASE)**.

Questo è il metodo per stabilire in modo sicuro le chiavi che entrambi i dispositivi potranno utilizzare per stabilire la comunicazione. In questo passaggio, il Commissario attiva anche l'accesso sicuro (fail-safe). Un fail-safe fornisce un modo per ripristinare il dispositivo allo stato originale qualora la messa in servizio non venga completata correttamente.

3. **Richiesta informazioni e DAC:** Il commissario esamina le informazioni contenute sull'endpoint 0 e su tutti gli altri endpoint presenti. Successivamente configura le informazioni sulle normative del commissionato utilizzando il comando `SetRegulatoryConfig`. Le informazioni sulle normative includono informazioni come la configurazione della posizione (per interni/esterni/entrambi) del dispositivo o l'impostazione del codice paese.
   L'obiettivo della procedura di attestazione del commissionato è determinare se un dispositivo è stato certificato e se è un dispositivo originale di tipo Matter. Il commissario estrae il certificato per l'attestazione del dispositivo (DAC) e il certificato Product Attestation Intermediate (PAI) dal commissionato. Tali certificati contengono l'ID fornitore, l'ID prodotto e la chiave pubblica di attestazione. Una volta ricevuti i certificati, viene inviata una richiesta di verifica che deve essere firmata dalla chiave privata di attestazione e che il commissario utilizza per stabilire l'autenticità del commissionato.
   Successivamente il commissario inoltre invia una richiesta di firma del certificato (CSR) al commissionato. Quest'ultimo crea una coppia di chiavi operative univoche che verrà utilizzata all'interno di **Certificate Authenticated Session Establishment (CASE)** in un secondo momento. Il commissionato restituisce le informazioni CSR risultanti al commissario.

4. **Aggiunta certificato NOC:** Il commissario utilizza le informazioni CSR ricevute dal commissionato e le trasmette all'Administrative Domain Manager (ADM) per generare un Node Operational Certificate (NOC) verificato. Il commissario installa successivamente il certificato radice sul commissionato utilizzando il comando `AddTrustedRootCertReq`, dopodichè installa il certificato operativo del nodo utilizzando il comando `AddNOC`.

5. **Configurazione della rete:** Il Commissario configura la rete operativa nella sede. Questo passaggio è necessario per i dispositivi Thread o Wi-Fi. Non è necessario per i dispositivi connessi tramite Ethernet poiché già connesso alla rete. Per farlo vengono usati comandi come `ScanNetworks`, `AddOrUpdateWifiNetwork` e `ConnectNetwork`.

Quando il nodo appena commissionato è connesso alla rete, il commissario utilizza il Rilevamento operativo per trovare il nodo sulla rete operativa. Il rilevamento operativo è il processo mediante il quale vengono trovati nodi commissionati nella rete operativa tramite DNS-SD. Se la Commissione è un dispositivo Wi-Fi, utilizzerà mDNS per rilevare il dispositivo. Tale funzione aiuta il commissario e altri nodi della rete a sapere quale indirizzo IP e porta utilizza il commissionato.

6. **Creazione sessione CASE:** Una volta individuato il nodo appena commissionato, viene stabilita una sessione CASE tra il commissario e il dispositivo. Questa sessione è avviata dal commissario e riceve risposta dal dispositivo. In questa fase, i certificati operativi vengono scambiati e viene stabilita una fiducia condivisa verificando che siano nella stessa struttura logica.

7. **Ultimazione della messa in servizio:** Il commissario usa CASE per inviare il comando `CommissioningComplete` al dispositivo appena commissionato. Questo è l'ultimo passaggio della procedura di messa in servizio.
   `CommissioningComplete` disattiva inoltre automaticamente il timer di sicurezza. Una volta completata la messa in servizio, il dispositivo opera come qualsiasi altro nodo sulla rete operativa.

Durante l'intera fase di commissioning inoltre, viene definito un ulteriore meccanismo di sicurezza che prevede l'utilizzo di una chiave di sicurezza temporanea (Temporary Security Key, TSK) per garantire la sicurezza della comunicazione tra il dispositivo e l'applicazione mobile. La TSK viene generata durante la fase di commissioning e viene utilizzata solo per la comunicazione tra il dispositivo e l'applicazione mobile durante la fase di configurazione iniziale. Una volta completata la fase di commissioning, la TSK viene sostituita da una chiave di sicurezza permanente (Permanent Security Key, PSK) che viene utilizzata per tutte le comunicazioni successive tra il dispositivo e la rete Matter.

3.2 Multi-Admin

Prima di introdurre l'ultima funzionalità di Matter introduciamo il concetto di ecosistema. Per ecosistema si intende un insieme di dispositivi connessi fra loro che dispongono di un unico Fabric comune. Comunicano attraverso i loro nodi e su di essi possono essere costruite ulteriori funzionalità.
Una delle funzionalità principali di Matter viene chiamata Multi-Admin.

Il Multi-Admin in Matter si riferisce alla capacità di avere più amministratori in un ecosistema di dispositivi IoT basato su questo protocollo. In pratica, un amministratore di un dispositivo può concedere l'accesso ad altri amministratori, dando loro la possibilità di gestire il dispositivo o l'intero sistema. Ciò consente una maggiore flessibilità nell'amministrazione del sistema, in quanto non c'è un unico punto di controllo.

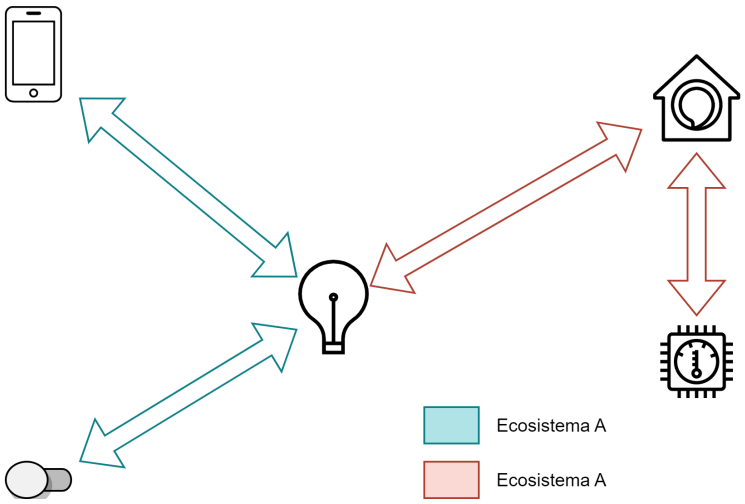

Figura 3.4: Rappresentazione della funzione Multi-Admin attraverso due ecosistemi [16]

Come si può vedere nell'immagine, questa lampadina fa parte di due ecosistemi distinti. Ciascuno di questi ecosistemi non conosce l'altro e sono alimentati da due organizzazioni separate (Ecosistema A ed Ecosistema B).

Tale funzionalità risulta utile in diversi contesti, ad esempio presa una famiglia ed una relativa abitazione, ogni familiare può avere accesso a diversi dispositivi al suo interno, questo può essere dovuto ad una questione di sicurezza (si pensi ad un bambino che può gestire solo le luci della sua camera) oppure anche ad una questione di livello organizzativo (si pensi ad un ospite e al fatto che non possa accedere al termostato per esempio).

Per poter aggiungere più amministratori nello stesso dispositivo è sufficiente richiedere un nuovo commissioning all'amministratore corrente. Una volta aggiunto un nuovo amministratore esso potrà svolgere azioni di lettura, scrittura e invocazione nei limiti concessi sui dispositivi. Ogni azione eseguita verrà quindi attribuita all'amministratore che richiama tale funzionalità e verrà registrato all'interno del dispositivo.
La presenza di più amministratori oltre che a migliorare la fruibilità dei dispositivi rappresenta anche una sicurezza aggiuntiva in quanto la responsabilità è divisa tra i diversi amministratori, riducendo il rischio di un singolo punto di fallimento.

3.3 Integrazione con Zigbee

Lo sviluppo di questo nuovo sistema tuttavia ha portato a diverse domande riguardo la compatibilità con altre tecnologie. Matter tuttavia è stato pensato anche in questo scenario, infatti è predisposto per il supporto di dispositivi esterni al suo protocollo.

Per poter comunicare con essi è necessaria la presenza di un dispositivo Bridge che consente la comunicazione fra diversi protocolli.

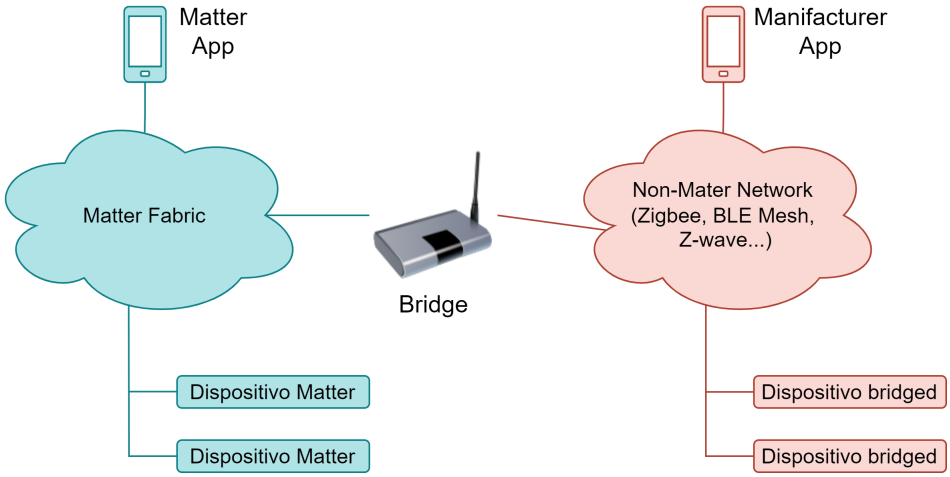

Figura 3.5: Rappresentazione di un bridge fra Matter e altri protocolli[6]

I dispositivi Non-Matter sono esposti come dispositivi Bridged ai nodi dell'ecosistema Matter. Il dispositivo Bridge esegue la traduzione tra Matter e altri protocolli in modo che i nodi Matter possano comunicare con i dispositivi Bridged.
Prendendo come esempio l'integrazione fra rete Matter e Zigbee andiamo ora a vedere come possono essere integrati. Di seguito si riporta un esempio che collega due luci Zigbee all'ecosistema Matter:

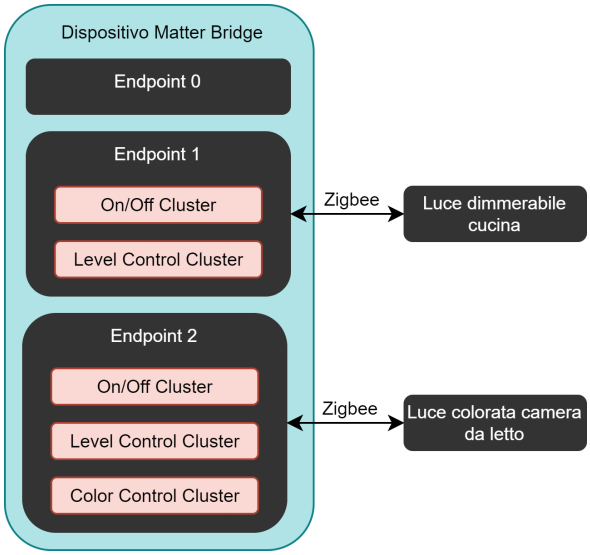

Figura 3.6: Matter-Zigbee Bridge[6]

Nella figura notiamo la presenza di due luci collegate mediante Zigbee. Tali luci sono collegate ad un Bridge Matter il quale associa diversi Cluster.
Andiamo adesso a vedere nel dettaglio come é composto il dispositivo Bridge.

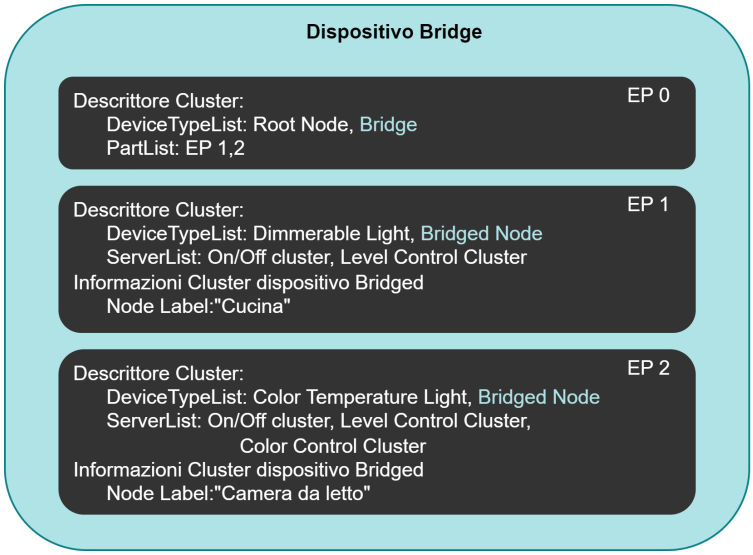

Figura 3.7: Composizione di un dispositivo Bridge[6]

Nell'endpoint 0, il tipo di dispositivo è definito come bridge. Il campo PartsList elenca tutti gli endpoint dei dispositivi bridged; ogni endpoint rappresenta un dispositivo sul lato non Matter del bridge.

Il Descrittore Cluster di ogni endpoint fornisce invece informazioni sul particolare dispositivo bridged.

Analizziamo ora come avviene l'integrazione e la comunicazione fra Matter e Zigbee.

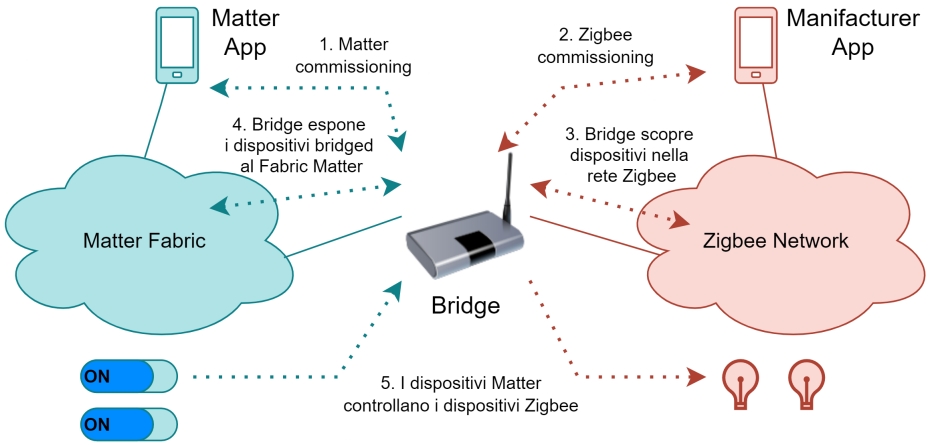

Figura 3.8: Esempio integrazione Matter con Zigbee[6]

Fase 1. Il bridge, un tipo di dispositivo definito in Matter, deve seguire il processo di messa in servizio standard di Matter per unirsi al Fabric Matter.

Fase 2. Anche il dispositivo Bridge Matter-Zigbee deve unirsi alla rete Zigbee. Diversamente da Matter, la specifica Zigbee non impone alcun processo di messa in servizio standard, lasciando ai fornitori di dispositivi la decisione sul flusso di lavoro per la distribuzione delle chiavi di collegamento. Il codice di installazione è il metodo più comune, Zigbee 3.0.

Fase 3. Una volta che il dispositivo Bridge si unisce alla rete Zigbee, scoprirà i dispositivi supportati nella rete Zigbee trasmettendo il comando `MatchDescriptorRequest`. Tale comando include il Profilo desiderato e i relativi Cluster associati. Per poter definire i Cluster il dispositivo effettuerà delle domande alla rete Zigbee. I dispositivi Zigbee corrispondenti risponderanno con un descrittore di corrispondenza (`MatchDescriptorResponse`) con il proprio indirizzo di rete. Per ogni luce Zigbee abbinata, il bridge aggiungerà un endpoint dinamico, in Matter, che sta per Bridged Zigbee Device.

Fase 4. Il bridge espone tutti i dispositivi bridged al Fabric Matter, che segue il metodo `OperationalDiscovery` definito dalle specifiche Matter.

Fase 5. Ora i controller nel fabric Matter possono controllare le luci nella rete Zigbee con l'aiuto del bridge.

Alcune note da tenere in considerazione:
Il metodo di interazione nei passaggi 2 e 3 è definito dai fornitori di dispositivi e dal protocollo stesso, non rientrando nell'ambito di Matter. Inoltre i dispositivi bridged possono essere aggiunti o rimossi dinamicamente dalla rete.
Tali concetti espressi in questi punti sono applicabili in modo analogo ad altre reti verso la quale si intende effettuare il bridging.

Conclusioni

Riepilogando quanto detto nei capitoli in precedenza, il protocollo Matter rappresenta una soluzione interessante per l'interoperabilità degli smart device e per la creazione di ecosistemi di prodotti IoT. Il suo focus sulla sicurezza, l'implementazione su diverse piattaforme e l'uso di tecnologie standardizzate come IPv6, 6LoWPAN e Thread, lo rendono un protocollo promettente per l'industria IoT.

Tra i punti salienti del protocollo Matter troviamo l'architettura basata su nodi, la gestione sicura dei dispositivi tramite la PKI, il commissioning standardizzato e semplificato, l'uso di ecosistemi multi-produttore e la flessibilità per l'integrazione di tecnologie wireless come Zigbee.

Per quanto riguarda i possibili scenari futuri, il protocollo Matter sta guadagnando sempre più adesioni da parte dei produttori di dispositivi IoT. Ciò potrebbe portare a una maggiore interoperabilità tra dispositivi di diverse marche, semplificando la vita degli utenti finali.

Dal punto di vista commerciale, il protocollo Matter potrebbe rappresentare una grande opportunità per le aziende che cercano di entrare nel mercato IoT, consentendo loro di concentrarsi sulla creazione di prodotti di alta qualità senza doversi preoccupare dell'implementazione di una piattaforma proprietaria.

Infine, in termini di sviluppo, il protocollo Matter potrebbe subire ulteriori evoluzioni, ad esempio nell'ambito dell'integrazione di tecnologie wireless come Wi-Fi7 o nel supporto di nuovi tipi di dispositivi IoT. In ogni caso, il suo approccio open-source e la partecipazione di un vasto ecosistema di sviluppatori renderanno possibile la creazione di nuove funzionalità e l'aggiornamento del protocollo in modo continuo.

Concludendo, il protocollo Matter sembra avere tutte le carte in regola per diventare uno standard importante nel mondo IoT, offrendo sicurezza, interoperabilità e flessibilità.

Bibliografia

[1] CSA Connectivity Standards Alliance. *Matter network transport*. URL: https://csa-iot.org/developer-resource/matter-network-transport/. (Ultimo accesso: 22-02-2023).

[2] CSA Connectivity Standards Alliance. *Matter Security & Privacy Webinar – April 2022*. URL: https://csa-iot.org/developer-resource/matter-security-webinar/. (Ultimo accesso: 25-02-2023).

[3] CSA Connectivity Standards Alliance. *Matter Security and Privacy Fundamentals*. URL: https://csa-iot.org/wp-content/uploads/2022/03/Matter_Security_and_Privacy_WP_March-2022.pdf. (Ultimo accesso: 25-02-2023).

[4] CSA Connectivity Standards Alliance. *Matter-Security-Privacy*. URL: https://csa-iot.org/wp-content/uploads/2022/06/Matter-Security-Privacy_one-pager.pdf. (Ultimo accesso: 25-02-2023).

[5] CSA Connectivity Standards Alliance. *Matter, the new Global Standard for the Smart Home, Debuts at the Amsterdam Launch Event*. URL: https://www.youtube.com/watch?v=eFgG8isgMUc&t=3688s. (Ultimo accesso: 24-02-2023).

[6] Shu Chen. *Matter: Bridge for Non-Matter Devices*. URL: https://blog.espressif.com/matter-bridge-for-non-matter-devices-d3b7f003a004. (Ultimo accesso: 27-02-2023).

[7] Shu Chen. *Matter: Distributed Compliance Ledger (DCL)*. URL: https://blog.espressif.com/matter-distributed-compliance-ledger-dcl-4013c2376e7. (Ultimo accesso: 26-02-2023).

[8] Sergio Donato. *Cos'è e come funziona Thread, il protocollo di comunicazione usato da Matter per la casa connessa*. URL: https://www.dday.it/redazione/43768/cose-e-come-funziona-thread-il-protocollo-di-comunicazione-usato-da-matter-per-la-casa-connessa. (Ultimo accesso: 20-02-2023).

[9] Google. *Commissioning*. URL: https://developers.home.google.com/matter/primer/commissioning?hl=en. (Ultimo accesso: 27-02-2023).

[10] Google. *Concetti sui modelli di interazione*. URL: https://developers.home.google.com/matter/primer/interaction-model?hl=it. (Ultimo accesso: 23-02-2023).

[11] Google. *Il modello dei dati del dispositivo*. URL: https://developers.home.google.com/matter/primer/device-data-model?hl=it. (Ultimo accesso: 23-02-2023).

[12] Guo Jiacheng. *Matter security model*. URL: https://blog.espressif.com/matter-security-model-37f806d3b0b2. (Ultimo accesso: 26-02-2023).

[13] Ramya Kanthi Polisetti. *Thread: A Low-Power Mesh Network Protocol for IoT*. URL: https://developer.qualcomm.com/blog/thread-low-power-mesh-network-protocol-iot. (Ultimo accesso: 24-02-2023).

[14] Slavikus. *Discover your Thread Network*. URL: https://www.evehome.com/en/blog/discover-your-thread-network. (Ultimo accesso: 20-02-2023).

[15] Kedar Sovani. *Matter: Clusters, Attributes, Commands*. URL: https://blog.espressif.com/matter-clusters-attributes-commands-82b8ec1640a0. (Ultimo accesso: 23-02-2023).

[16] Kedar Sovani. *Matter: Multi-Admin, Identifiers, and Fabrics*. URL: https://blog.espressif.com/matter-multi-admin-identifiers-and-fabrics-a291371af365. (Ultimo accesso: 27-02-2023).

[17] Infineon Technologies. *How does Matter work for consumers?* URL: https://www.infineon.com/cms/en/product/promopages/matter/#How-Matter-works. (Ultimo accesso: 27-02-2023).

Ringraziamenti

Eccoci arrivati alla parte finale, i Ringraziamenti.
Questi ultimi anni sono trascorsi in fretta, tuttavia ogni attimo è risultato indispensabile per poter essere qui adesso. Sono stati anni pieni di momenti importanti, felici e tristi, che hanno condizionato il mio percorso. Sono stati anni in cui ho conosciuto tantissime persone con cui ho condiviso tantissime conoscenze e senza la quale non sarei qui ora probabilmente.

Ma passiamo ora ai ringraziamenti, prima di tutto ringrazio i miei genitori e i miei fratelli, per avermi permesso di studiare e per avermi supportato durante i periodi più stressanti. Ringrazio i miei amici più stretti, Nicolò e Alex per essermi stati sempre vicini facendomi distrarre dalle sessioni infinite. Ringrazio Alessia, per avermi accompagnato in questo percorso come collega e soprattutto come amica. Ringrazio Paola e Alberto, seppur la nostra amicizia è iniziata da poco tempo non avete esitato un attimo per prestarmi il vostro aiuto. Grazie poi a tutte le persone che mi sono state vicine e che mi hanno aiutato nei momenti difficili. Ringrazio tutti i docenti del corso ed in particolare il prof. Callegati, senza la quale non avrei potuto svolgere questa tesi.

Infine ci tengo a ringraziare la persona che più di tutte mi è stata vicina, Martina. A lei devo veramente tanto, è stata in grado di sopportarmi e supportarmi tutti questi anni, senza mai perdere la fiducia in me e senza farmi pesare tutte le occasioni perse per via dello studio. Insieme abbiamo passato tantissimi momenti, felici e tristi, e da ogni evento ne siamo sempre usciti insieme, perciò grazie per essermi stata vicina anche in questa esperienza.

Ma ora è tempo di guardare avanti, la fine di questo percorso è solo l'inizio di nuove strade che sono certo faranno di me una persona più matura.