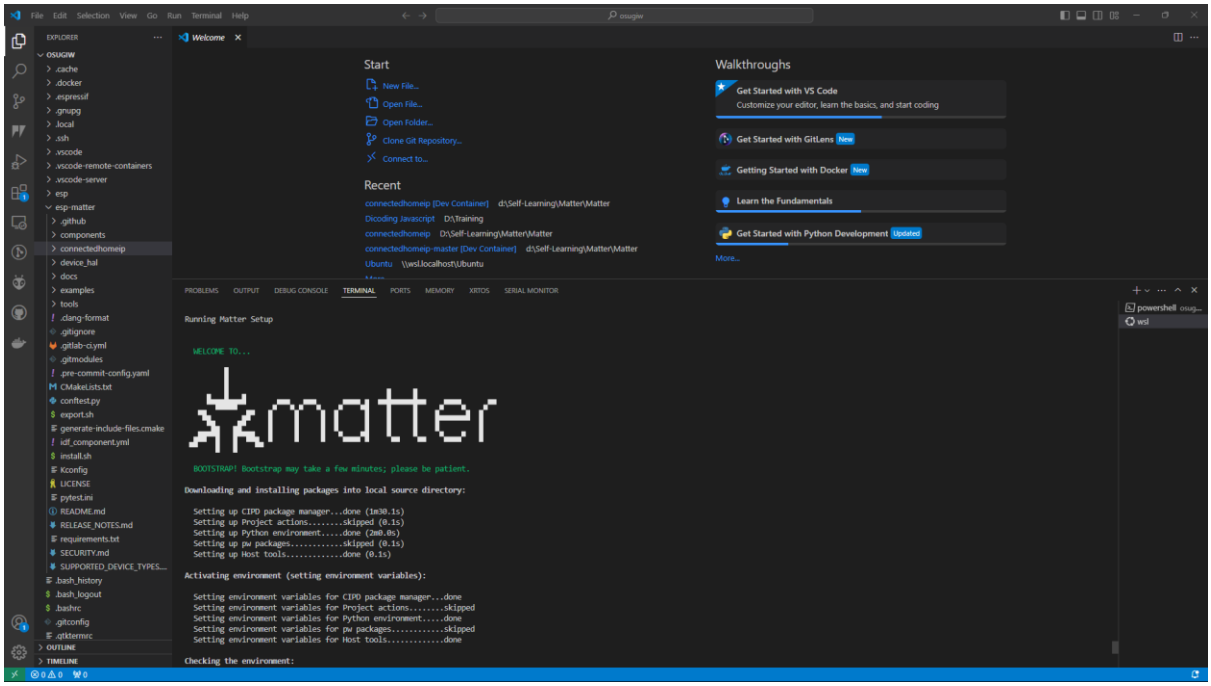
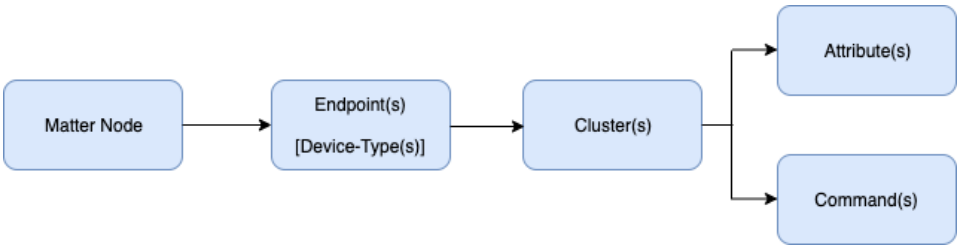
Matter



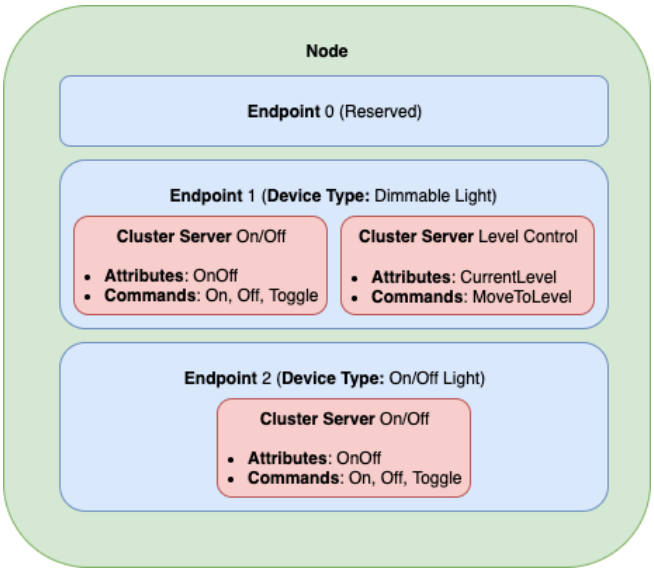
Data Model



- **Node** – Unique network address that expose some functionality. Commonly, it refers to physical device.
- **Endpoint** – It is a virtual device that provides services that could be logically grouped together. Each Matter Node can have several endpoints. The Matter specification defines certain common Device Types. While endpoint 0 is reserved, it contains certain services that are applicable to the entire node. It also known as *root node* device type and the clusters inside this endpoint include:
    - Basic Information Cluster Server – Information including node, firmware version, manufacturer, etc.
    - ACL Cluster Server – Allow configuration of the Access Control Lists for this node.

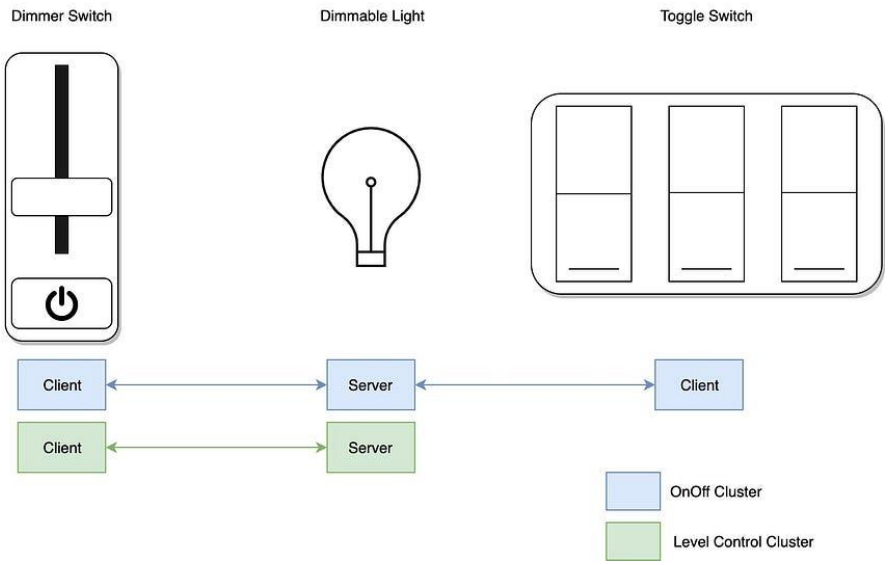
- Network Commissioning Cluster Server – Allows configuration of a network (Wi-Fi, Ethernet, Thread) on the node.

- **Clusters** – A group that has common function. An endpoint can have several clusters.
- **Attributes** – Parameters that can be read or written to.
- **Commands** – Set of command that can be used for triggering a specific behaviour.



Cluster Servers and Clients

In every Matter cluster has a **Cluster server** and a **Cluester Client counterpart**.

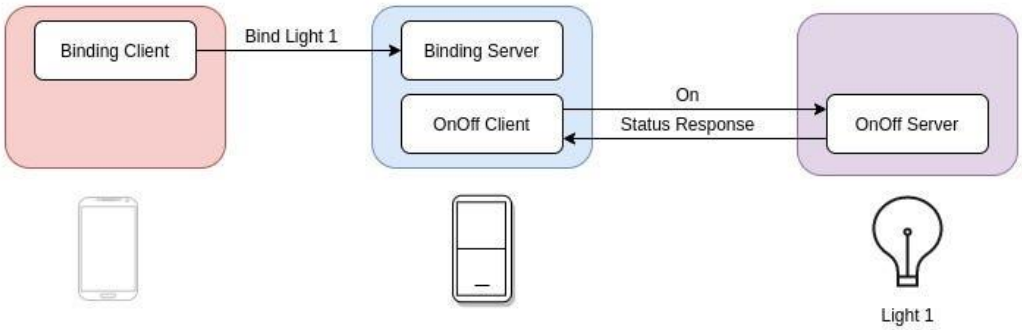


Device-to-Device Automations

As the Matter Cluster has servers and clients, they can be configured to communicate with a specific device when certain conditions is met. This can be achieved through **Device Binding,** a Matter phone application can establish binding between devices, even though they are not from same vendors.

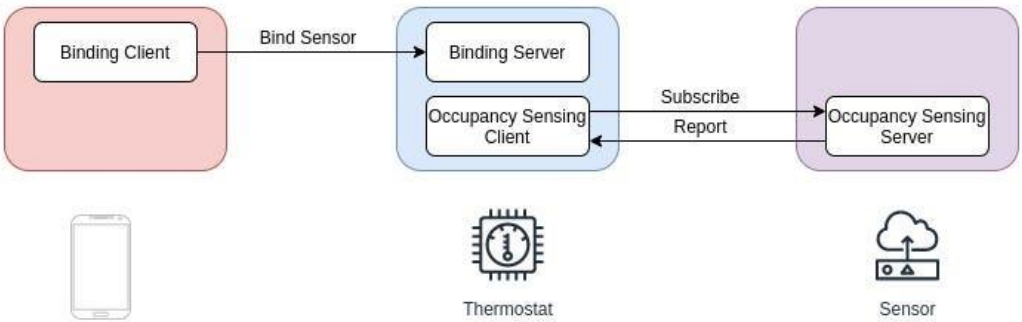
1. Synchronous Control

   It requires a Binding cluster server that offers the binding service.

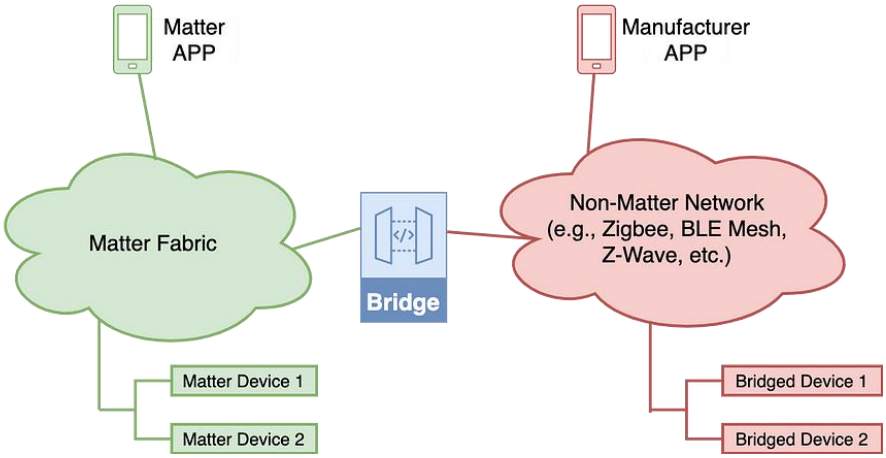


2. Asynchronous Notification (Subscribe-Report)

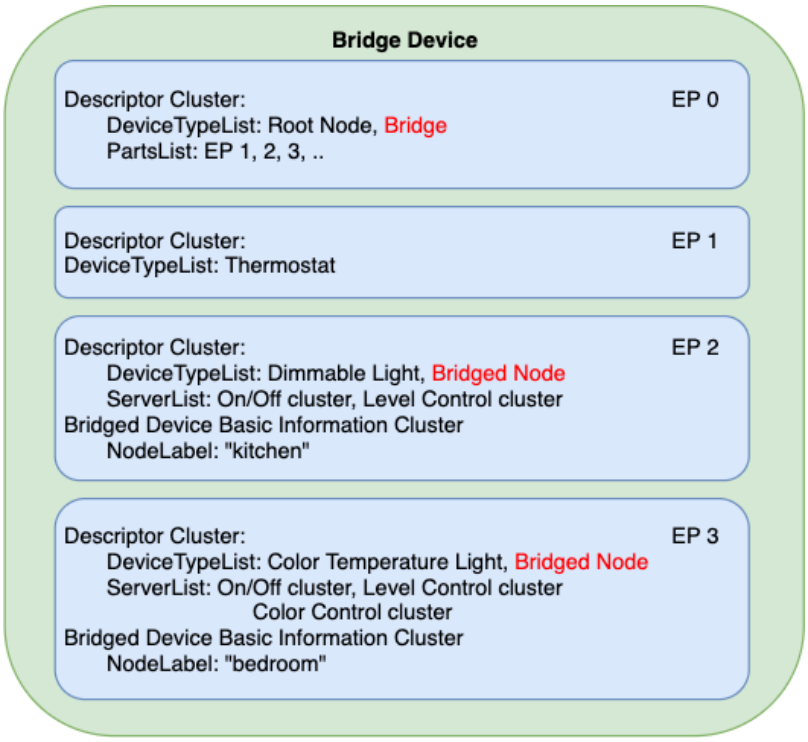
   It has two roles: subscriber and publisher, a device can subscribe to attributes and/or events (every single change) on the publisher. After successfully bind, the publisher will periodically send data to the subscriber.



Matter Bridge between Matter and non-Matter Device



A Bridge allows non-Matter device (BLE, Wi-Fi, Zigbee, etc.) to join the Matter ecosystem (Matter Fabric) and communicate with Matter devices. A bridge device has some additional data parameters on each cluster, namely **Descriptor**. A descriptor on endpoint 0 has different information, it has *PartList Field* that tells all the endpoints for bridged devices and each endpoint in non-Matter devices represent one device. The native Matter functionality may also be implemented on a bridge device that have multiple communication protocols (i.e. Wi-Fi and 802.15.4.).



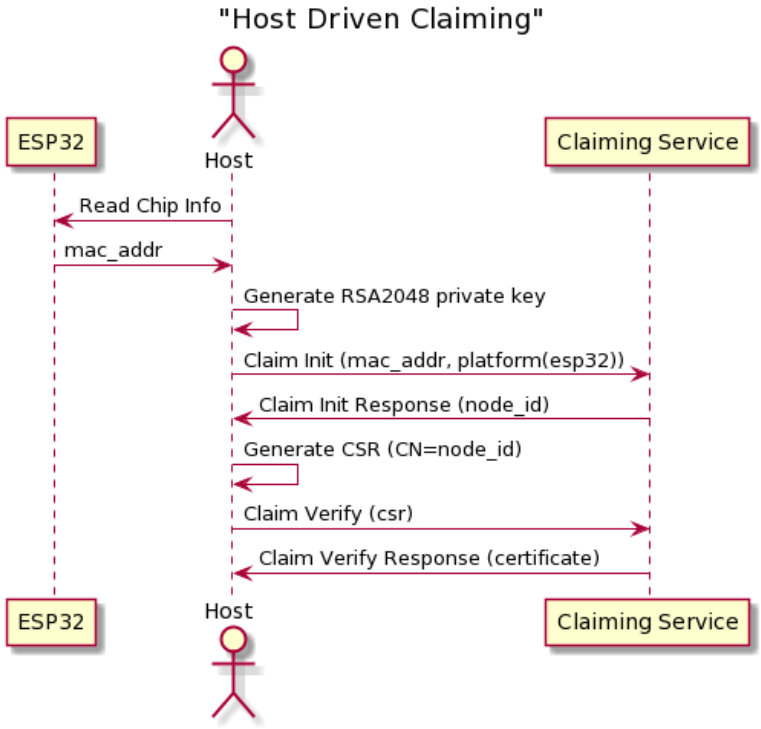
Here is the example on how to control a Zigbee device using Matter protocol:

1. The Bridge device follow the standard Matter commissioning process to join the Matter fabric.
2. The Matter-Zigbee Bridge Device join the Zigbee network.

3. Once the Bridge Device joins the Zigbee network, the Matter-Zigbee Bridge device can discover the supported devices in the Zigbee network by broadcasting the **Match Descriptor request** command (including: desired profile, in- and out-clusters). Then the corresponding Zigbee devices will reply the **Match Descriptor Response** with its network address included.
4. The Bridge exposes all the Bridged Devices to the Matter Fabric.
5. The controllers in the Matter fabric can control the lights in the Zigbee network with the help of Bridge.

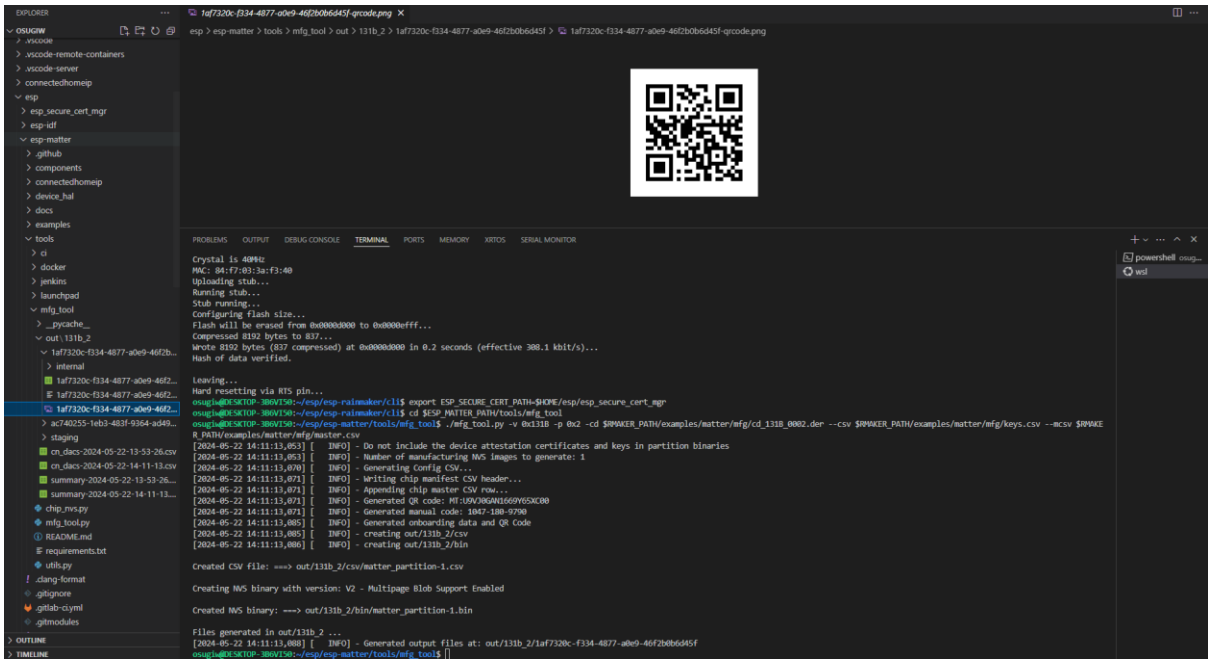
Rainmaker

Claiming device certificates using **Host driven claiming**.



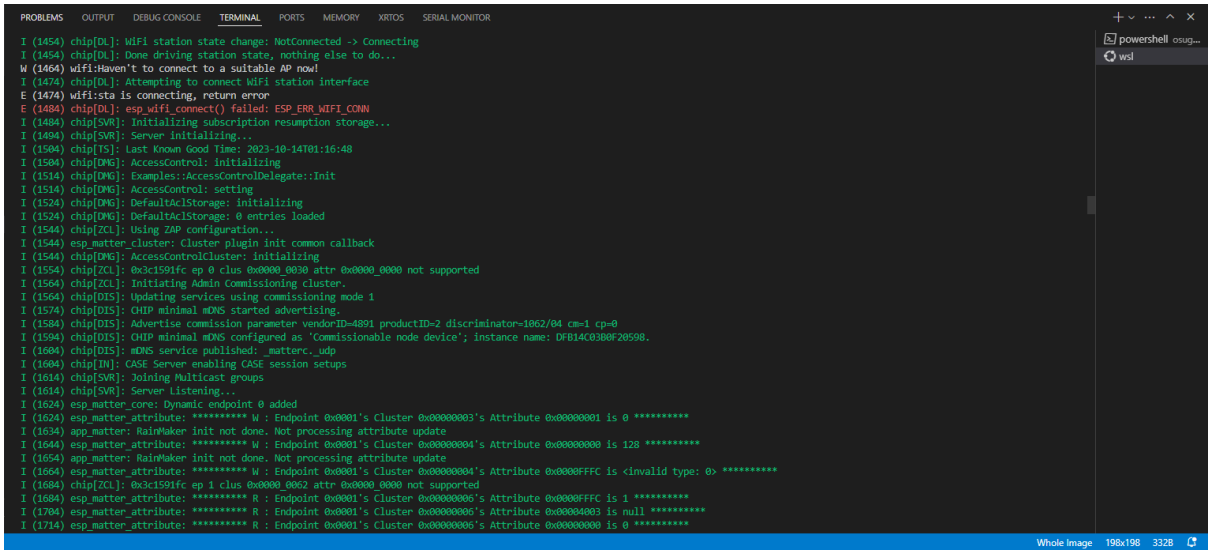

```
osugiw@DESKTOP-3B6VI50:~/esp/esp-rainmaker/cli$ ./rainmaker.py claim --matter /dev/ttyUSB0

Claiming process started. This may take time.
Claim initiate started
Claim initiate done
Generating CSR
Claim verify done
Claim certificate received

Saving claiming data info at location:  /home/osugiw/.espressif/rainmaker/claim_data/GitHub_NbUwC26NwsHmzgTfj2mMbD/84F7033AF340/
Claiming done
Time(s):5.075908422470093
dev_cert tlv: total length = 512
ca_cert tlv: total length = 464
tlv header bytes = 0
priv_key tlv: total length = 144
Total length of tlv data = 1120
Generated esp_secure_cert partition: /home/osugiw/.espressif/rainmaker/claim_data/GitHub_NbUwC26NwsHmzgTfj2mMbD/84F7033AF340/esp_secure_cert.bin
Flashing binary onto node
esptool.py v4.7.0
Serial port /dev/ttyUSB0
Connecting....
Detecting chip type... ESP32-C3
Chip is ESP32-C3 (QFN32) (revision v0.3)
Features: WiFi, BLE, Embedded Flash 4MB (XMC)
Crystal is 40MHz
MAC: 84:f7:03:3a:f3:40
Uploading stub...
Running stub...
Stub running...
Configuring flash size...
Flash will be erased from 0x0000d000 to 0x0000efff...
Compressed 8192 bytes to 837...
Wrote 8192 bytes (837 compressed) at 0x0000d000 in 0.2 seconds (effective 308.1 kbit/s)...
Hash of data verified.
```

Generating the factory NVS binary using *mfg_tool* of esp_matter SDK.



The factory binary generated above should be flashed onto the fctry partition (default : 0x3fa000 for ESP32-C6 and 0x3e0000 for other chips).

After being connected to the same WiFi network then the device will bring up the Matter configurations.



Create the Matter environments (Matter Fabric)

```
I (17874) chip[FP]: Validating NOC chain
I (18074) chip[FP]: NOC chain validation successful
I (18074) chip[FP]: Added new fabric at index: 0x1
I (18074) chip[FP]: Assigned compressed fabric ID: 0xD0EA88A83186780F, node ID: 0x87F0F22AE9FBE713
I (18084) chip[TS]: Last Known Good Time: 2023-10-14T01:16:48
I (18094) chip[TS]: New proposed Last Known Good Time: 2024-05-15T00:00:00
I (18104) chip[TS]: Updating pending Last Known Good Time to 2024-05-15T00:00:00
I (18114) chip[ZCL]: OpCreds: ACL entry created for Fabric index 0x1 CASE Admin Subject 0xFFFFFFFD00010001
I (18114) chip[DIS]: Advertise operational node D0EA88A83186780F-87F0F22AE9FBE713
I (18124) chip[DIS]: CHIP minimal mDNS configured as 'Operational device'; instance name: D0EA88A83186780F-87F0F22AE9FBE713.
I (18144) chip[DIS]: mDNS service published: _matter._tcp
I (18144) chip[ZCL]: OpCreds: successfully created fabric index 0x1 via AddNOC
I (18154) chip[EM]: <<< [E:54308r S:21651 M:197009518 (Ack:20495583)] (S) Msg TX to 1:FFFFFFFB00000000 [780F] [UDP:[FE80::903C:63FF:FE08:D870%st1]:49473] --- Type 0001:09 (IM:InvokeCommandRes
ponse)
I (18174) app_matter: Fabric is updated
I (18234) chip[EM]: >>> [E:54308r S:21651 M:20495584 (Ack:197009518)] (S) Msg RX from 1:FFFFFFFB00000000 [780F] --- Type 0000:10 (SecureChannel:StandaloneAck)
I (18994) chip[EM]: >>> [E:54309r S:0 M:148747509] (U) Msg RX from 0:18FEB33028C686EF [0000] --- Type 0000:30 (SecureChannel:CASE_Sigma1)
I (18994) chip[IN]: CASE Server received Sigma1 message . Starting handshake. EC 0x3fc9ca68
I (19004) chip[EM]: <<< [E:54309r S:0 M:254010696 (Ack:148747509)] (U) Msg TX to 0:0000000000000000 [0000] [UDP:[FE80::903C:63FF:FE08:D870%st1]:49473] --- Type 0000:10 (SecureChannel:Standalo
neAck)
I (19024) chip[SC]: Received Sigma1 msg
I (19034) chip[SC]: CASE matched destination ID: fabricIndex 1, NodeID 0x87F0F22AE9FBE713
I (19284) chip[EM]: <<< [E:54309r S:0 M:254010697 (Ack:148747509)] (U) Msg TX to 0:0000000000000000 [0000] [UDP:[FE80::903C:63FF:FE08:D870%st1]:49473] --- Type 0000:31 (SecureChannel:CASE_Sig
ma2)
```

Commission the Matter device using the ESP-Rainmaker application by scanning the generated QR Code.

```
I (22154) chip[EM]: >>> [E:54314r S:21652 M:255446558 (Ack:181111769)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0000:10 (SecureChannel:StandaloneAck)
I (22204) chip[EM]: Retransmitting MessageCounter:181111772 on exchange 54315r Send Cnt 1
I (22254) chip[EM]: >>> [E:54315r S:21652 M:255446559 (Ack:181111772)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0000:10 (SecureChannel:StandaloneAck)
I (22264) chip[EM]: >>> [E:54315r S:21652 M:255446560 (Ack:181111772)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0000:10 (SecureChannel:StandaloneAck)
I (22274) chip[EM]: >>> [E:54316r S:21652 M:255446561] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0001:0a (IM:TimedRequest)
I (22294) chip[EM]: <<< [E:54316r S:21652 M:181111773 (Ack:255446561)] (S) Msg TX to 1:C16DF896049A85DA [780F] [UDP:[FE80::903C:63FF:FE08:D870%st1]:49473] --- Type 0001:01 (IM:StatusResponse)
I (22424) chip[EM]: >>> [E:54316r S:21652 M:255446562 (Ack:181111773)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0001:08 (IM:InvokeCommandRequest)
I (22434) esp_matter_command: Received command 0x00000000 for endpoint 0x0000's cluster 0x0000003C
I (22444) chip[ZCL]: Received command to open commissioning window
I (22444) chip[DIS]: Updating services using commissioning mode 2
I (22454) chip[DIS]: CHIP minimal mDNS started advertising.
I (22464) chip[DIS]: Advertise operational node D0EA88A83186780F-87F0F22AE9FBE713
I (22464) chip[DIS]: CHIP minimal mDNS configured as 'Operational device'; instance name: D0EA88A83186780F-87F0F22AE9FBE713.
I (22484) chip[DIS]: mDNS service published: _matter._tcp
I (22484) chip[DIS]: Advertise commission parameter vendorID=4891 productID=2 discriminator=0316/01 cm=2 cp=0
I (22494) chip[DIS]: CHIP minimal mDNS configured as 'Commissionable node device'; instance name: 310417C55282EBCD.
I (22514) chip[DIS]: mDNS service published: _matterc._udp
I (22514) chip[ZCL]: Commissioning window is now open
I (22524) chip[EM]: <<< [E:54316r S:21652 M:181111774 (Ack:255446562)] (S) Msg TX to 1:C16DF896049A85DA [780F] [UDP:[FE80::903C:63FF:FE08:D870%st1]:49473] --- Type 0001:09 (IM:InvokeCommandRe
sponse)
I (22544) app_matter: Commissioning window opened
I (22544) chip[DL]: BLE deinit successful and memory reclaimed
I (22894) chip[EM]: Retransmitting MessageCounter:181111774 on exchange 54316r Send Cnt 1
I (22894) chip[EM]: >>> [E:54316r S:21652 M:255446562 (Ack:181111773)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0001:08 (IM:InvokeCommandRequest)
I (22904) chip[EM]: <<< [E:54316r S:21652 M:181111775 (Ack:255446562)] (S) Msg TX to 1:C16DF896049A85DA [780F] [UDP:[FE80::903C:63FF:FE08:D870%st1]:49473] --- Type 0000:10 (SecureChannel:Stan
daloneAck)
I (22924) chip[EM]: >>> [E:54316r S:21652 M:255446563 (Ack:181111774)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0000:10 (SecureChannel:StandaloneAck)
I (22934) chip[EM]: >>> [E:54316r S:21652 M:255446564 (Ack:181111774)] (S) Msg RX from 1:C16DF896049A85DA [780F] --- Type 0000:10 (SecureChannel:StandaloneAck)
I (28254) chip[EM]: >>> [E:53792r S:0 M:205580650] (U) Msg RX from 0:306C7716E00782D0 [0000] --- Type 0000:20 (SecureChannel:PBKDFParamRequest)
I (28254) chip[EM]: <<< [E:53792r S:0 M:254010700 (Ack:205580650)] (U) Msg TX to 0:0000000000000000 [0000] [UDP:[FE80::903C:63FF:FE08:D870%st1]:51953] --- Type 0000:10 (SecureChannel:Standalo
neAck)
I (28274) chip[EM]: <<< [E:53792r S:0 M:254010701 (Ack:205580650)] (U) Msg TX to 0:0000000000000000 [0000] [UDP:[FE80::903C:63FF:FE08:D870%st1]:51953] --- Type 0000:21 (SecureChannel:PBKDFPar
amResponse)
I (28294) chip[SVR]: Commissioning session establishment step started
I (28564) chip[EM]: >>> [E:53792r S:0 M:205580650] (U) Msg RX from 0:306C7716E00782D0 [0000] --- Type 0000:20 (SecureChannel:PBKDFParamRequest)
I (28564) chip[EM]: <<< [E:53792r S:0 M:254010702 (Ack:205580650)] (U) Msg TX to 0:0000000000000000 [0000] [UDP:[FE80::903C:63FF:FE08:D870%st1]:51953] --- Type 0000:10 (SecureChannel:Standalo
neAck)
```

References

[1] esp-rainmaker/examples/matter at master · espressif/esp-rainmaker (github.com)