

# Tock and Thread Networking

**Tyler Potyondy**, UC San Diego  
tpotyondy@ucsd.edu

# What is Thread Networking?

Thread is a low power, secure and future-proof mesh networking technology for IoT products.



## BUILT FOR IOT

Low power, secure and robust wireless mesh built on IP



## CONVERGENCE & COEXISTANCE

IP as a point of convergence



## GLOBAL SOLUTION

Open standard for smart homes and buildings



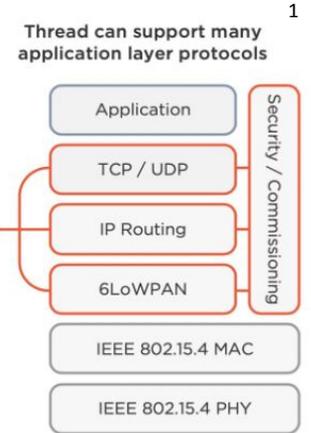
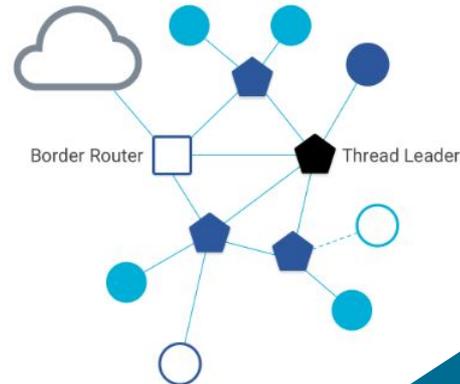
## FLEXIBLE & FUTURE PROOF

Enabling interoperability

 Apple revolutionized personal technology with the introduction of the Macintosh in 1976. Today, Apple leads the w... <a href="#">Visit Site →</a>	 Our mission is to organize the world's information and make it universally accessible and useful. <a href="#">Visit Site →</a>	 Inventronics is a premier worldwide producer of lighting products, including LED drivers, sensors, control system... <a href="#">Visit Site →</a>	 Amazon is guided by four principles: customer obsession rather than competitor focus, passion for invention, comm... <a href="#">Visit Site →</a>
 Founded in 1961, Lutron Electronics is headquartered in Coopersburg, Pennsylvania. From dimmers for the home, to... <a href="#">Visit Site →</a>	 Nordic Semiconductor is specializing in ultra-low power short-range wireless communication supporting Bluetooth®... <a href="#">Visit Site →</a>	 NXP Semiconductors N.V. (NASDAQ: NXP) enables a smarter, safer and more sustainable world through innovation. As... <a href="#">Visit Site →</a>	 Qualcomm is the world's leading wireless technology innovator and the driving force behind the development, launc... <a href="#">Visit Site →</a>
 SmartThings is Samsung Electronics' premier IoT technology, focused on enabling and reimagining connected living... <a href="#">Visit Site →</a>	 Siemens is a global powerhouse focusing on the areas of electrification, automation and digitalization. One of th... <a href="#">Visit Site →</a>	 Silicon Labs (NASDAQ: SLAB) is a leader in secure, intelligent wireless technology for a more connected world. Du... <a href="#">Visit Site →</a>	 Somfy's leading smart management solutions for homes and buildings have been improving people's daily lives for o... <a href="#">Visit Site →</a>

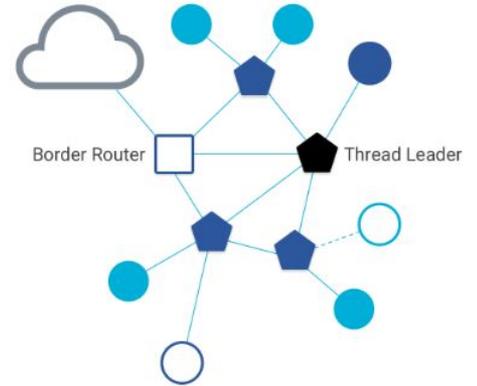
# What is Thread Networking?

- Thread builds a networking layer on top of 802.15.4
- Utilizes IPv6 routing for each node, allowing for interoperability with existing networks
- Combines star and mesh topologies to provide a robust network

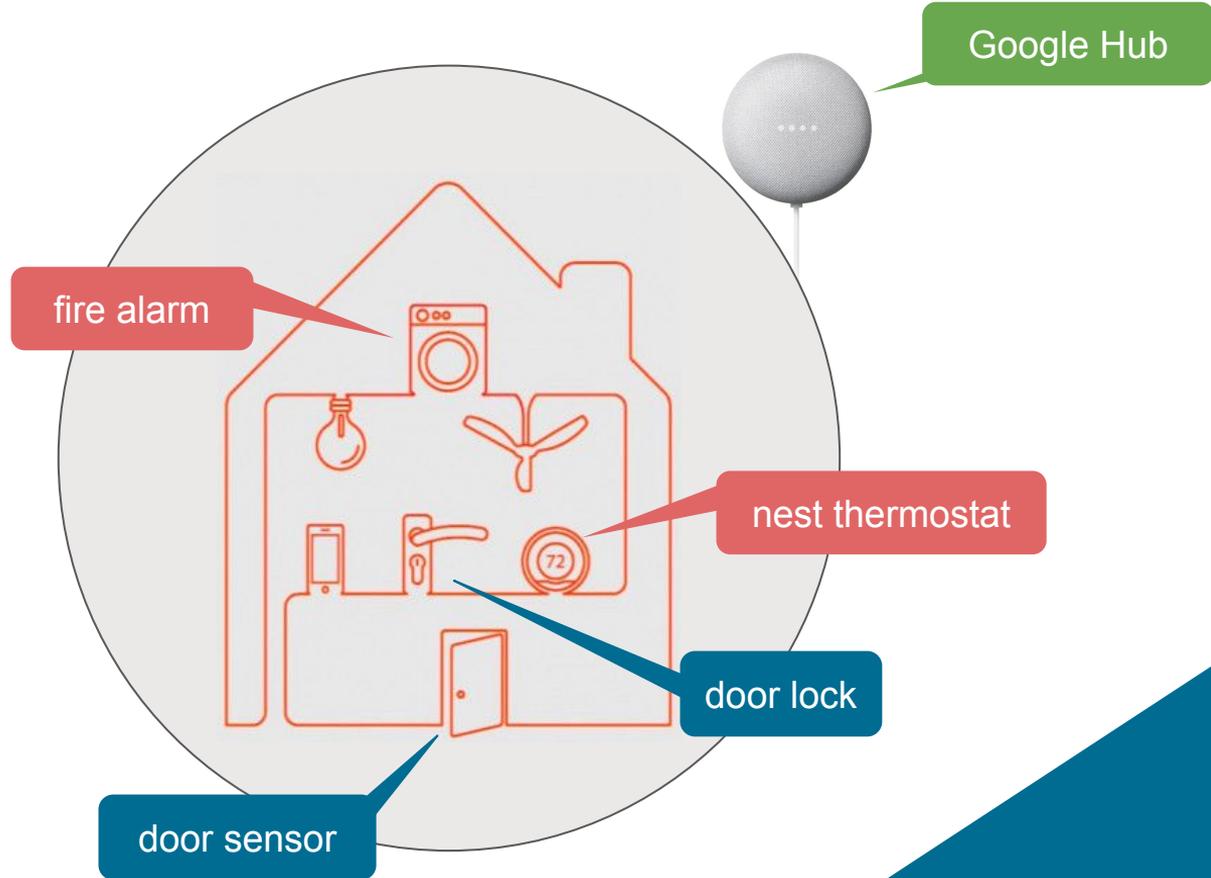


# Thread Networking - Node Types

- **Border Router**
  - Node providing bridge from Thread network to external internet
- **Router**
  - Responsible for maintaining network topology
  - Forwards packets for network devices
  - Mesh communication with other routers
  - Power intensive node
- **Child**
  - Communicates with one router
  - Does not forward packets
  - Can disable transceiver to enter low power state (sleepy end device)



# Example of Real World Thread Network



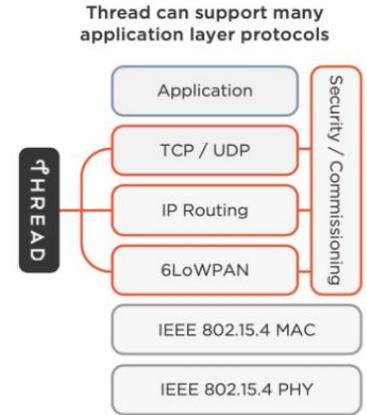
Border Router

Router Node

Child Node

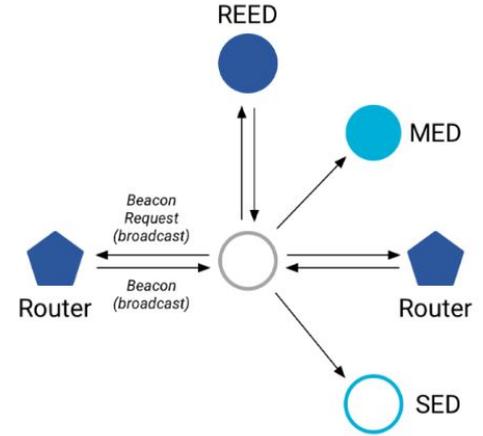
# Thread Networking and Tock

- Tock currently supports each networking layer theoretically required for deploying a Thread network
- Tock implements the AES128-CCM encryption needed to secure Thread payloads
- Can we add kernel support for Thread networking?



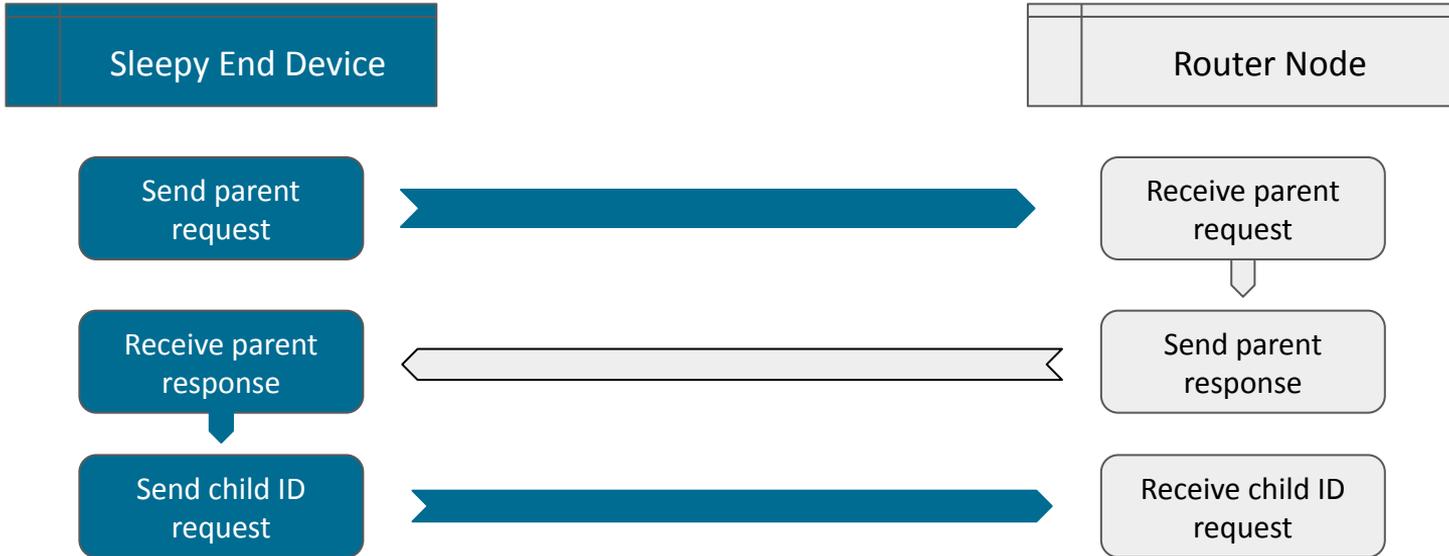
# Tock – Sleepy End Device

- A sleepy end device (SED) is the simplest of Thread node types
- OpenThread is Google's open source implementation of the Thread networking specification
- **Goal:** Tock kernel level support to successfully join an OpenThread network



OPENTHREAD  
released by Google

# Tock – Sleepy End Device



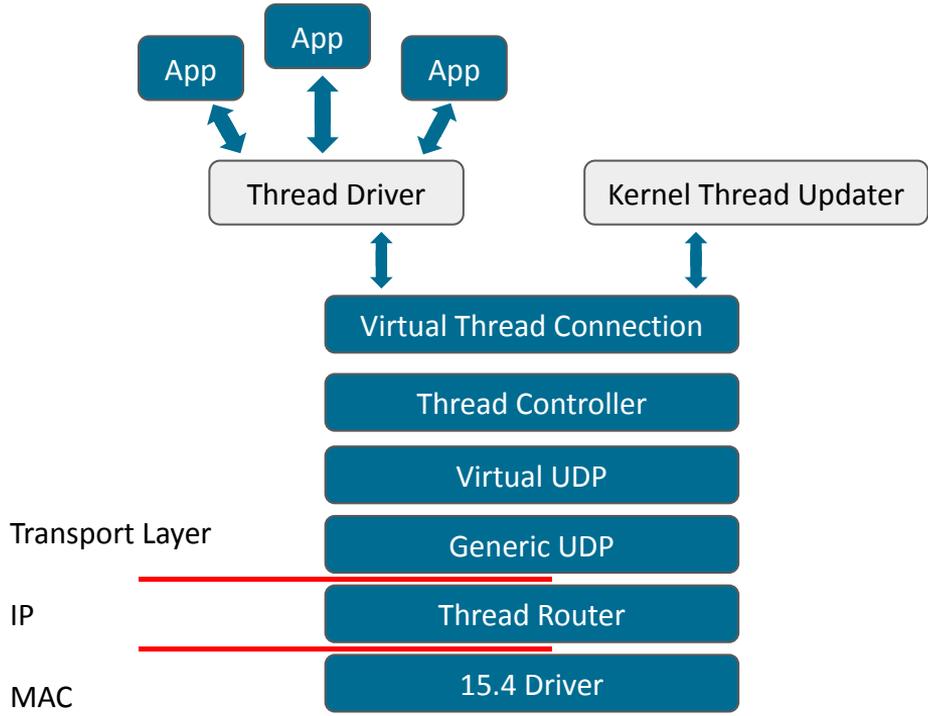
# Tock – Sleepy End Device

Successfully performed “handshakes” required for OpenThread to recognize a Tock SED as a node in the Thread network

Successfully proved that the Tock networking stack and AES128-CCM encryption can support a Thread network

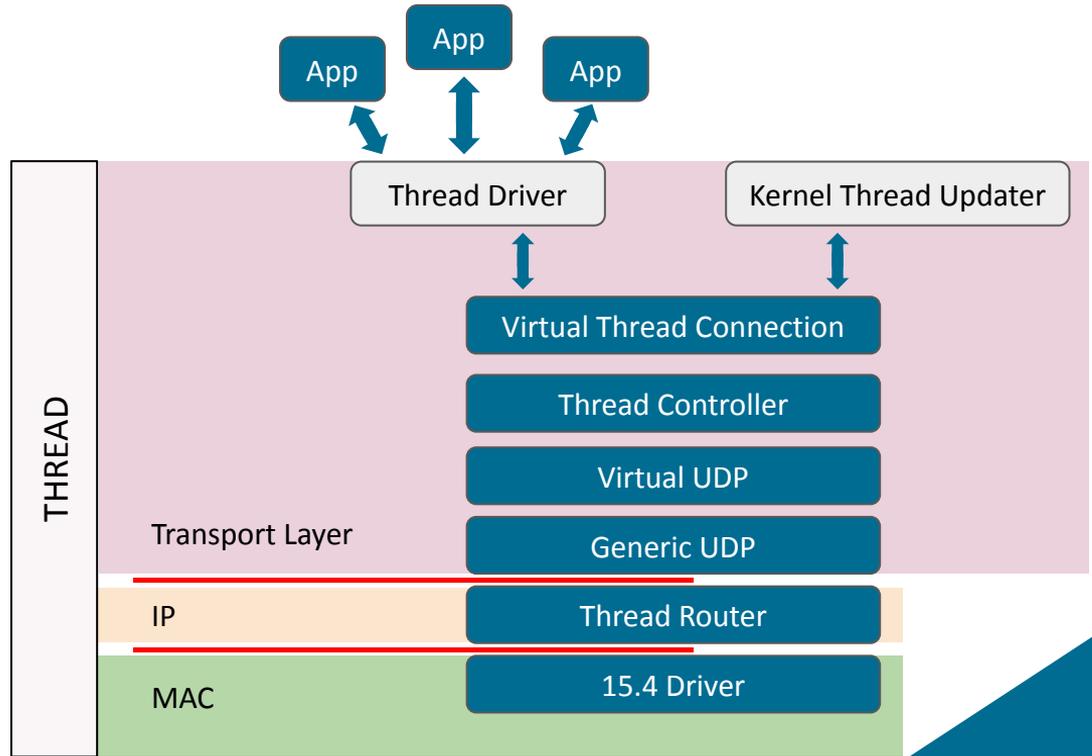
# Open Questions/Current Work

How to best design the Thread Networking Capsule to be generalizable?



# Open Questions/Current Work

**Unique Challenge:**  
Thread spans 3 layer of  
the networking stack



# Open Questions/Current Work

How to best track the device's Thread network state?

Will the kernel's sharing of cryptographic resources lead to starvation and subsequently dropped packets?

Should each device support multiple distinct Thread networks?

How to best incorporate Thread across the networking stack?

Should all apps be allowed to access the Thread Network?  
Should we possess a means to differentiate access?

# Questions?